

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-016332

(43)Date of publication of application : 19.01.1996

(51)Int.Cl.

G06F 3/12

B41J 5/30

G06F 13/00

(21)Application number : 06-150777

(71)Applicant : CANON INC

(22)Date of filing : 01.07.1994

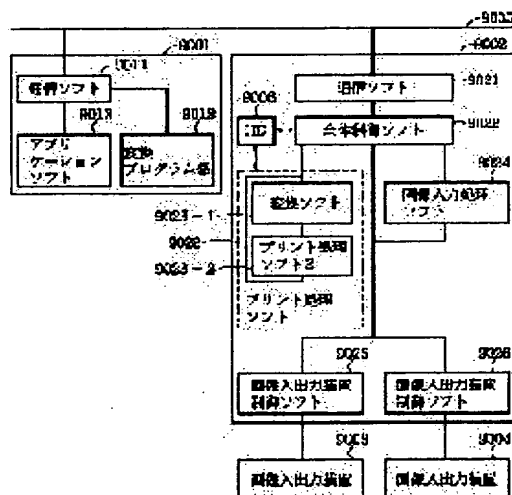
(72)Inventor : YOKOMIZO YOSHIKAZU
 HASHIMOTO HIROHIKO
 YAMADA OSAMU
 TODA YUKARI
 SAITO KAZUHIRO
 TODA MASANARI
 NEGISHI SAKURIKI
 FUKUDA YASUO
 SUGIYAMA MITSUMASA
 SHISHIZUKA JIYUNICHI
 TAKAOKA MAKOTO
 KOBAYASHI SHIGETADA
 MITA YOSHINOBU
 SUGIURA SUSUMU

(54) SCANNER PRINTER SERVER SYSTEM AND DATA PROCESSING METHOD FOR THE SAME

(57)Abstract:

PURPOSE: To keep a speedy data processing environment by preventing the data throughput of each host from being lowered by considerably reducing the transfer processing burden of each host when transferring printing data from each host to a scanner printer server.

CONSTITUTION: When a transform program file for transforming the printing data provided with data structure corresponding to the operating system of each host to the printing data corresponding to a page description language in a scanner printer server device is transferred from a communication software 9011 of each host to the scanner printer server device, based on each transferred transform program file, a print processing software 9023 transforms the printing data transferred from each host to the printing data corresponding to the page description language in the scanner printer server device.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the
 examiner's decision of rejection or application
 converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-16332

(43)公開日 平成8年(1996)1月19日

(51)Int.Cl. ⁹	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 3/12	A			
	C			
B 4 1 J 5/30	Z			
G 0 6 F 13/00	3 5 7 Z	7368-5E		

審査請求 未請求 請求項の数10 O L (全 80 頁)

(21)出願番号 特願平6-150777

(22)出願日 平成6年(1994)7月1日

(71)出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72)発明者 横溝 良和

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 橋本 裕彦

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(72)発明者 山田 修

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

(74)代理人 弁理士 小林 将高

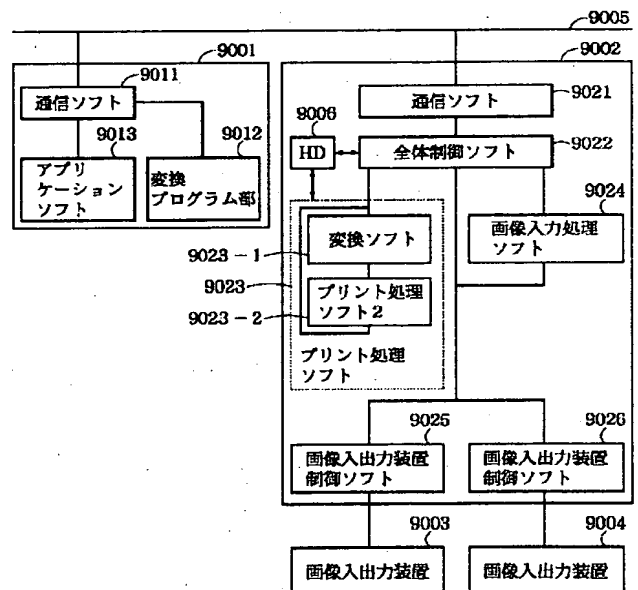
最終頁に続く

(54)【発明の名称】 スキャナプリンタサーバシステムおよびそのデータ処理方法

(57)【要約】

【目的】 各ホストからスキャナプリンタサーバ装置への印刷データ転送時における各ホストの転送処理負担を格段に軽減して、各ホストのデータ処理能力低下を防止して快速なデータ処理環境を維持できる。

【構成】 各ホストの通信ソフト9011から各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルがスキャナプリンタサーバ装置に転送されたら、該転送された各変換プログラムファイルに基づいてプリント処理ソフト9023が各ホストから転送される印刷データをスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する構成を特徴とする。



9001 ホストマシン

9002 スキャナプリンタサーバ

9005 ネットワーク

【特許請求の範囲】

【請求項 1】 スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバ装置とから構成されるスキャナプリンタサーバシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段を各ホストに設け、各ホストの転送手段から転送される各変換プログラムファイルに基づいてスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する変換手段をスキャナプリンタサーバ装置に設けたことを特徴とするスキャナプリンタサーバシステム。

【請求項 2】 各ホストは、スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを複数記憶する記憶手段を設け、転送手段が起動しているアプリケーションのデータ構造に基づいて前記記憶手段から選択した変換プログラムファイルを転送することを特徴とする請求項 1 記載のスキャナプリンタサーバシステム。

【請求項 3】 転送手段は、印刷要求毎に印刷データとともに変換プログラムファイルを転送することを特徴とする請求項 1 記載のスキャナプリンタサーバシステム。

【請求項 4】 転送手段は、ネットワーク初期設定時に、記憶手段に記憶されたいずれかの変換プログラムファイルを転送することを特徴とする請求項 2 記載のスキャナプリンタサーバシステム。

【請求項 5】 転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を確認することを特徴とする請求項 1 記載のスキャナプリンタサーバシステム。

【請求項 6】 転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無をスキャナプリンタサーバ装置に問い合わせ確認することを特徴とする請求項 1 記載のスキャナプリンタサーバシステム。

【請求項 7】 転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を転送履歴情報に基づいて確認することを特徴とする請求項 1 記載のスキャナプリンタサーバシステム。

【請求項 8】 スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバ装置とから構成されるスキャナプリンタサーバシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段を各ホストに設け、各ホストの転送手段から転送される各変換プログラムファイルに基づいてスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する変換手段をスキャナプリンタサーバ装置に設けたことを特徴とするスキャナプリンタサーバシステム。

ヤナプリンタサーバシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段を各ホストに設け、各ホストの転送手段から転送された変換プログラムファイルを記憶する記憶手段と、前記転送手段から転送される印刷データを解析して前記記憶手段に対応する変換プログラムファイルが登録されているかどうかを判定する判定手段と、この判定手段による判定結果に基づいて解析された印刷データをスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求する変換ファイル転送要求手段と、各ホストの転送手段から転送される印刷データを記憶手段に記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する変換手段とをスキャナプリンタサーバ装置に設けたことを特徴とするスキャナプリンタサーバシステム。

【請求項 9】 変換手段は、解析された印刷データの形式に従って記憶手段に登録されたいずれかの変換プログラムファイルを選択してスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換することを特徴とする請求項 8 記載のスキャナプリンタサーバシステム。

【請求項 10】 スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバ装置とから構成されるスキャナプリンタサーバシステムのデータ処理方法において、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルをいずれかのホストからスキャナプリンタサーバ装置に転送する転送工程を有し、スキャナプリンタサーバ装置が各ホストから転送される各変換プログラムファイルに基づいて各ホストから転送された印刷データをスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換することを特徴とするスキャナプリンタサーバシステムのデータ処理方法。

【発明の詳細な説明】**【0001】**

【産業上の利用分野】 本発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介して複数のホストと通信可能に接続されたスキャナプリンタサーバ装置を有するスキャナプリンタサーバ

ーシステムおよびそのデータ処理方法に関するものである。

【0002】

【従来の技術】従来、この種の入出力機器、特にプリンタは、セントロニクスと呼ぶパラレルインタフェースか、RS-232Cと呼ばれるシリアルインタフェースを介してコンピュータに直接接続されることが多かった。

【0003】また、スキャナ（原稿読み取り装置）は、標準でないビデオインタフェースか、SCSIインタフェースでコンピュータに直接接続されることが多かった。このような、インタフェースでは、快適な利用環境を整えるためには、1台1台のコンピュータにプリンタとスキャナを接続せざるを得ず、資源の効率的利用が図れなかった。すべてのコンピュータにプリンタとスキャナを接続することは、現実的には不可能で、従来は、フロッピーディスクを持って他のコンピュータを借に行くか、キャラクタベースのプリンタのプリンタならば、セントロニクスの分岐ボックスを共同利用することが多かった。

【0004】近年、ネットワーク技術の進展により、LANインタフェースを有するプリンタがいくつか販売されるようになってきた。LANを介してコンピュータとプリンタを接続することにより、ネットワーク上のどのコンピュータからでも手軽に印刷が可能となっている。

【0005】これらのネットワークプリンタは、LANのインタフェースに加え、米国のアドビ社が開発したポストスクリプト（PostScript）というページ記述言語を搭載した例が大部分であり、一種の業界標準を形成している。

【0006】また、従来の通信プロトコルの構造を複数のレイヤに分けて、各レイヤ毎にパケットの形状を独立に定め、各レイヤ毎にタスク（プロセス）を割り当てて、マルチタスク（マルチプロセス）でプログラムを構築することにより、各レイヤの機能を明確化するとともに、独立設計を容易としている。また、下位のレイヤは上位のレイヤに対するサービスエンティティの提供に専念させることができる等の利点があった。

【0007】さらに、単一のスキャナやプリンタは、ホストコンピュータと直接接続された形態で使用されてきた。また、一体型のスキャナプリンタは、一般的には単体で複写機として使用されているので、このようなスキャナやプリンタをネットワークに接続し、スキャナプリンタサーバーとして機能させる形態では使用されていなかった。

【0008】一方、ネットワーク上でプリンタサーバーに接続し、プリンタを共有する構成があった。通常、このようなプリンタサーバーでは、ホストコンピュータからプリンタの仕様に合せてデータをプリンタに送っていた。

【0009】また、ネットワーク上で、該ネットワークに接続されたリモートホストよりスキャンを行うネットワークスキャナという使用法はなく、スキャナ依頼を行うホストコンピュータに画像を取り込む場合には、そのホストコンピュータ自身に接続されたスキャナより読み込むか、あるいはネットワーク上にある他のホストコンピュータに接続されたスキャナより読み込み、その画像をネットワークを経由して依頼を行ったホストに転送する方法がある。また、スキャナで入力した画像を他の文字、図形情報とともに印刷する場合、通常は印刷依頼を行うホストコンピュータ上でスキャナ画像、文字、図形情報を合成し、ネットワークに接続されたプリンタで出力を行っていた。

【0010】また、従来この種のサーバー装置の形式は大きく分けて下記のような種類があった。

【0011】1.ホストコンピュータから専用インタフェースを通じプリンタに接続され、ホストコンピュータ内でビットイメージに展開しプリンタに伝送する。ネットワーク対応は、ホストコンピュータが面倒を見る。主にワークステーション程度の高機能ホストコンピュータをベースにした形式。

【0012】2.ホストコンピュータからはページ記述言語が伝送され、プリンタ側で持つインタープリタ機能によりビットイメージに変換する方式。ネットワークプリンタと称されるもの。

【0013】3.ホストコンピュータから伝送されるページ記述言語を専用のビットイメージに変換する部分を通じ、プリンタに専用インタフェースを通じ画像データを伝送するもの。

【0014】さらに、従来のネットワーク上に存在するサーバー装置のダウンロード方法は、下記（a）、

（b）による場合が多い。

【0015】（a）サーバー装置を立ち上げる時に、必要なプログラムをダウンロードホストからダウンロードを行う。

【0016】（b）プリンタ等のプリント処理を依頼するホストコンピュータがフォント、フィルタ等のダウンロードを行う。

【0017】また、従来のネットワーク環境のパラメータ管理方式としては、ネットワーク上のホスト、ユーザの管理に行うものに、UNIX環境で広く使用されるNIS（Network Information System）がある。このNISにおいては、ネットワーク上のあるドメイン内にマスターサーバー装置、スレーブサーバー装置、クライアントを設定し、ドメイン内のホスト、ユーザ管理等のパラメータは、通常マスターサーバーで管理されている。また、マスターサーバーの変更情報は、スレーブサーバーに自動的に転送される。このため、マスターサーバー装置、スレーブサーバー装置を意識せずにパラメータ情報の参照を行うことができる。

【0018】

【発明が解決しようとする課題】しかしながら、上記ページ記述言語を使用してシステムを構築して、ネットワークに共有使用可能なプリンタを接続する場合、すなわちネットワークプリンタインタフェースを標準化して、どのメーカーのプリンタとも、自由に接続できるようにする場合、2つの原則がある。

【0019】1. アプリケーションソフトは、各プリンタの言語や制御方法に依存してはならない。

【0020】2. プリンタは、各アプリケーションの出力するコードの違いを気にしてはならない。

【0021】このように、アプリケーションソフトとプリンタとの間に、両者の使用の違いを吸収する仕組み、いわゆるプリンタドライバ（プログラム）が存在しなければならない。

【0022】しかるに、上記ポストスクリプトを使用する場合、アプリケーションがポストスクリプトコードを出力し、プリンタがポストスクリプトコードを理解するから正常に印刷できるわけであるが、逆に言えば、アプリケーションソフトは、プリンタの言語であるポストスクリプトに依存しており、プリンタは、アプリケーションの出力するポストスクリプトコードに依存していると言える。

【0023】また、ポストスクリプトはプリンタ用であって、スキャナには使用できない。

【0024】このように、上記ポストスクリプトの使用の制約から、スキャナとプリンタとを備えるスキャナプリンタ（複写装置）をネットワークの共有資源として、各ホストコンピュータからのスキャナ要求、プリンタ要求を処理する場合、アプリケーションに制約が生じ、スキャナプリンタのスペックを最大限に発揮させることができなくなる等の問題点があった。

【0025】さらに、上記のように通信ネットワーク上のプロトコルをレイヤ構造とすると、本来のネットワーク間通信では使用しない、タスク（プロセス）間通信が増大するためにCPUの処理負担が増大しデータ処理効率が低下する。また、各タスク毎に設定するバッファメモリが増大して、メモリ資源を浪費する等の問題点があった。

【0026】また、最近では、ネットワーク化が進んでおり、インテリジェントビル等の全体にLAN（ローカル・エリア・ネットワーク）を張り巡らせた大規模なネットワーク化が進んでいる。さらには、WAN（ワイドエリアネットワーク）のような、公衆回線でLANを直結した全国規模のネットワーク、さらにはISDN等の高度情報網が整備されつつある。

【0027】このため、ビルの別のフロアや、別のビル、あるいは東京から大阪のホストコンピュータを利用すること等が可能となりつつある。従って、プリンタサーバーでは、従来までの比較的狭い範囲の利用に留まら

ず、非常に広域な利用が可能となった。

【0028】また、スキャナやプリンタも高機能化、高解像度化、カラー化が進んでいる。このため、様々な形式でのスキャナからの読み取り、プリンタへの出力が可能となってきている。従来のホストコンピュータに直結したスキャナやプリンタは、ホストコンピュータがそれぞれの設定や画像データの入出力を行っていた。インタフェース（I/F）、画像データ形式、通信速度等は、1対1の関係であり、システムで独自で構築しても何ら問題はなかった。

【0029】一方、ネットワーク上にSPサーバーシステムを介して接続されたスキャナ、プリンタは、複数のホストコンピュータで利用できるため、資源の共有化が図れる上、異機種であるホストコンピュータに共通の利用環境を提供できるといった利点があり、効率的である。

【0030】しかしながら、ネットワークで共有化する故、下記のような新たな問題も発生する。

【0031】第1に、利用時間もまちまちであったり、同時に利用することもあり、それに伴って、例えば利用者AがSPサーバーシステムを介してプリンタを利用している最中に、利用者Bがプリンタ利用の要求を出した場合、SPサーバーが同様に振る舞うかが問題となる。

【0032】第2に、利用者Aが同様に、プリンタを利用している最中に、何らかのエラーが発生した場合、利用者Aにどのように、通知するか、また、その際、利用者Bが利用しようとしても、利用できないことを、どのように通知するかが問題となる。

【0033】第3に、SPサーバーシステムに異常が発生した場合、あるいはホストコンピュータ側に異常が発生した場合、お互いの通信が不能となり、どちらかがコマンド待ちの状態となり、システムがデッドロック状態になってしまう。そこで、SPサーバーシステム側もホストコンピュータ側もそのような状態で不良となることを避けるため、タイマ監視機能を備える場合がある。この場合、お互いのコマンドに対するレスポンスをある一定時間待ち、その時間を越えるような場合、再送要求を何回か発行して、それでも応答がない場合は自動的に通信を打ち切って初期状態に戻す動作を行う。

【0034】しかしながら、このような通信制御によれば、以下のような新たな問題が発生する。

【0035】ネットワークが大規模化して、1つのネットワークにゲートウェイあるいはルーターを設けて、他のネットワークと結ばれている場合であって、スキャナやプリンタの接続されたSPサーバーシステムとホストコンピュータが異なるネットワークにある場合の利用の時、どちらか一方のネットワークが遅いと、お互いコマンドに対するレスポンスが正常な場合であっても、タイマ監視機能で設定した時間を越えてしまうことが発生する。このような状況では、ゲートウェイあるいはルータ

一の遅延の大きな場合や、一方のネットワークが異常に混雑している場合にも発生する。

【0036】このような状況が発生した場合、ネットワークの通信遅延が大きいだけで、正常な通信を行っている場合であっても、S/Pサーバシステムやホストコンピュータのタイマ監視機能が作動し、自動的に再送要求を発行してしまい、ひいては自動的に初期状態に戻ってしまう。これにより、正常な通信であっても、利用者がスキャナやプリンタの資源を利用できないといった問題点があった。

【0037】さらに、従来のサーバ装置におけるデータダウンロード方式、パラメータ管理方式では、プログラム、フィルタ、フォント、パラメータ等のダウンロードおよびその管理は、ネットワーク上のドメインの中で、ある特定のサーバ装置から見た場合、上述のように共通の手段（上述ではマスタとスレーブが個別に処理を行う）によっては行われていないため、例えばスキャナプリンタをネットワークに接続して、各クライアントがスキャナ要求、プリンタ要求を処理するサーバ装置のダウンロード、パラメータ管理が問題となる。すなわち、ネットワーク上にスキャナ／プリンタ（S/P）サーバ装置を設置した場合、従来のデータダウンロード方式、パラメータ管理方式では、S/Pサーバ装置がプログラム、フィルタ、フォント、パラメータを一元管理できず、予期しないシステムダウンに即応できなくなる場合がある。

【0038】また、複数のダウンロードホスト中に、マスタホスト、スレーブホストを設定した場合、スレーブホスト同士での通信は行われておらず、マスタホストが何らかの原因でダウンした場合、パラメータ等の変更ができなくなり、システムサービスが行えなくなる等の幾多の問題点があった。

【0039】さらに、クライアント側でアプリケーションソフトを起動している最中に、印刷要求処理を並行処理する際、クライアント側の処理能力が低いと、CPUのレスポンスが極めて低下し、クライアント側のアプリケーション処理速度が低下し、所望のアプリケーションソフトを快適に実行させることができないというネットワークプリント処理特有の問題点があった。

【0040】本発明は、上記の問題点を解決するためになされたもので、本発明に係る第1～第10の発明の目的は、ネットワーク上の各ホストからスキャナプリンタサーバ装置に転送される所定のデータ構造を有する各ホストの印刷データをスキャナプリンタサーバ装置内のページ記述言語に変換する変換プログラムファイルを各ホストから転送して取得し、該取得した変換プログラムを実行して各ホストの印刷データをスキャナプリンタサーバ装置内のページ記述言語に変換することにより、各ホストからスキャナプリンタサーバ装置への印刷データ転送時における各ホストの転送処理負担を格段に軽

減して、各ホストのデータ処理能力低下を防止して高速なデータ処理環境を維持できるスキャナプリンタサーバシステムおよびその通信処理方法を提供することである。

【0041】

【課題を解決するための手段】本発明に係る第1の発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバ装置とから構成されるスキャナプリンタサーバシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段を各ホストに設け、各ホストの転送手段から転送される各変換プログラムファイルに基づいてスキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する変換手段をスキャナプリンタサーバ装置に設けたものである。

【0042】本発明に係る第2の発明は、各ホストは、スキャナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを複数記憶する記憶手段を設け、転送手段が起動しているアプリケーションのデータ構造に基づいて前記記憶手段から選択した変換プログラムファイルを転送するように構成したものである。

【0043】本発明に係る第3の発明は、転送手段は、印刷要求毎に印刷データとともに変換プログラムファイルを転送するように構成したものである。

【0044】本発明に係る第4の発明は、転送手段は、ネットワーク初期設定時に、記憶手段に記憶されたいずれかの変換プログラムファイルを転送するように構成したものである。

【0045】本発明に係る第5の発明は、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を確認するように構成したものである。

【0046】本発明に係る第6の発明は、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無をスキャナプリンタサーバ装置に問い合わせ確認するように構成したものである。

【0047】本発明に係る第7の発明は、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を転送履歴情報に基づいて確認するように構成したものである。

【0048】本発明に係る第8の発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネッ

トワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバー装置とから構成されるスキャナプリンタサーバーシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段を各ホストに設け、各ホストの転送手段から転送された変換プログラムファイルを記憶する記憶手段と、前記転送手段から転送される印刷データを解析して前記記憶手段に対応する変換プログラムファイルが登録されているかどうかを判定する判定手段と、この判定手段による判定結果に基づいて解析された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求する変換ファイル転送要求手段と、各ホストの転送手段から転送される印刷データを記憶手段に記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換する変換手段とをスキャナプリンタサーバー装置に設けたものである。

【0049】本発明に係る第9の発明は、変換手段は、解析された印刷データの形式に従って記憶手段に登録されたいずれかの変換プログラムファイルを選択してスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するように構成したものである。

【0050】本発明に係る第10の発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器が所定のインタフェースを介して接続され、かつ所定のネットワークを介してオペレーティングシステムが複数のホストと通信可能に接続されたスキャナプリンタサーバー装置とから構成されるスキャナプリンタサーバーシステムの通信処理方法において、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルをいずれかのホストからスキャナプリンタサーバー装置に転送する転送工程を有し、スキャナプリンタサーバー装置が各ホストから転送される各変換プログラムファイルに基づいて各ホストから転送された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するように構成したものである。

【0051】

【作用】第1の発明においては、各ホストの転送手段から各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置本体内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルがスキャナプ

リンタサーバー装置に転送されたら、該転送された各変換プログラムファイルに基づいて変換手段が各ホストから転送される印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換して、各ホスト側によるスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データへの変換処理をスキャナプリンタサーバー装置側で実行して、各ホストによる印刷データ転送処理負担を軽減する。

【0052】第2の発明においては、転送手段が起動しているアプリケーションのデータ構造に基づいて記憶手段から選択した変換プログラムファイルを転送して、各アプリケーションに対応した最適な変換プログラムファイルをスキャナプリンタサーバー装置に転送する。

【0053】第3の発明においては、転送手段は、印刷要求毎に印刷データとともに変換プログラムファイルを転送して、印刷要求時に対応する変換プログラムファイルを印刷データと対として確実に転送する。

【0054】第4の発明においては、転送手段は、ネットワーク初期設定時に、記憶手段に記憶されたいずれかの変換プログラムファイルを転送して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0055】第5の発明においては、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を確認して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0056】第6の発明においては、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無をスキャナプリンタサーバー装置に問い合わせ確認して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0057】第7の発明においては、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を転送履歴情報に基づいて確認して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0058】第8の発明においては、各ホストの転送手段から転送された変換プログラムファイルを記憶手段に記憶しておき、判定手段が各ホストの転送手段から転送される印刷データを解析して前記記憶手段に対応する変換プログラムファイルが登録されているかどうかを判定し、該判定結果に基づいて変換ファイル転送要求手段が解析された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求し、各ホストの転送手段から転送される印刷データを記憶手段に記憶し、各ホストから転送される印刷データを変換手段が記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換し

て、各ホストからの印刷要求時におけるスキナプリンタサーバ装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮する。

【0059】第9の発明においては、変換手段は、解析された印刷データの形式に従って記憶手段に登録されたいずれかの変換プログラムファイルを選択してスキナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換して、各ホストからの印刷要求時におけるスキナプリンタサーバ装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮する。

【0060】第10の発明においては、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルをいずれかのホストからスキナプリンタサーバ装置に転送し、スキナプリンタサーバ装置が各ホストから転送される各変換プログラムファイルに基づいて各ホストから転送された印刷データをスキナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換する。

【0061】

【実施例】図1は本発明の第1実施例を示すスキナプリンタサーバシステムの概要を説明するシステムブロック図である。なお、詳細なシステムブロック図は後述する。

【0062】図において、スキナ／プリンタ（SP）サーバは後述する。本システムは、大別してメインCPU回路1、イーサネット回路2、JPEG圧縮回路3、スキナ／プリンタインタフェース回路4より構成されている。なお、スキナ／プリンタ（SP）サーバに複数のスキナ／プリンタを接続する場合には、その台数分のスキナ／プリンタインタフェース回路を実装する。その台数の増減を容易にする為にスキナ／プリンタインタフェース回路は独立したボード上に構成されており、本体とは後述するVMEバス16で接続する。残りのメインCPU回路1、イーサネット回路2、JPEG圧縮回路3は1枚のボード上に構成されており、互いにローカルなバス14で結合されている。

【0063】4-1は第1のスキナ／プリンタインタフェース回路（第1のSPインタフェース回路）で、例えば商品名CLC-500等のデジタルカラー複写装置100（レーザビームカラー複写装置）とVMEバス16とのI/Oをインタフェースする。

【0064】4-2は第2のプリンタ／スキナインタフェース（第2のSPインタフェース回路）で、例えば商品名ピクセルジェット等のデジタルカラー複写装置200（バブルジェットカラー複写装置）とVMEバス16とのI/Oをインタフェースする。

【0065】メインCPU回路1は例えばR3000（商品名）等のCPU5とメモリ6等から成り、ここで

OSである、例えば VxWorks（商品名）の管理下ですべてのプログラムが実行される。メモリ6の補助記憶装置として、ハードディスク7がある。8はSCSIインタフェースである。イーサネット回路2はイーサネットトランシーバ9とデータをメモリ6にDMA転送する為のDMAコントローラ10等とから成る。イーサネットトランシーバ9はイーサネット（商品名）ローカルエリアネットワーク（LAN）11に接続する。JPEG圧縮回路3はJPEG圧縮／伸長コントローラ12とFIFOメモリ13等とから成る。メインCPU回路1、イーサネット回路2、JPEG圧縮回路3を互いに結合しているローカルバス14はVMEバストランシーバ15-1を介してVMEバス16に接続されている。

【0066】第1のSPインタフェース回路4-1はページメモリ17と第1の制御回路18等からなる。1ページ分のメモリが必要な理由は、デジタルカラー複写装置100が印刷／スキャンを開始すると途中でフロー制御する事ができないからである。ページメモリ17は、メモリの価格が高いため圧縮メモリを構成している。すなわち、このメモリ17に書き込まれるデータは、メモリ6でラスタライズされた画像のみである事に着目し、メモリ17は非可逆圧縮ではあるが、高圧縮率が得られるADCT圧縮方式を採用している。そのADCT圧縮チップはこの図では示していないが、前述のJPEG圧縮／伸長コントローラ12と同じチップを用いている。

【0067】ただし、このページメモリ17にADCT圧縮方式を用いる場合、圧縮データはここだけの閉じた回路で使用されるだけで、外部には出て行かないから、標準に則る必要はなく、実際、JPEGを改良した効率的な方法を用いている。第1の制御回路18は例えばZ-80（商品名）等のCPU等で構成し、複写装置100との画像以外の制御信号のやり取りをしている。ここのインタフェースの本数を減らすため、制御コマンドをシリアルで送っている。第1のSPインタフェース回路4-1はVMEバストランシーバ15-2を介してVMEバス16と接続している。

【0068】第2のSPインタフェース回路4-2はデータ縦横変換回路19、バッファメモリ20、画像処理回路21及び第2の制御回路22等とから成り。VMEバストランシーバ15-3を介してVMEバス16と接続している。データ縦横変換回路19は、プリント時は走査方向の画像のデータ形式を走査方向に対して90°に変換し、スキャン時は走査方向に対して90°のデータ形式を走査方向に変換する。バッファメモリ20はそのための変換用バッファとして用いる。

【0069】画像処理回路21は複写装置200に欠けている画像処理機能を補う為に入れている。第2の制御回路22は第1の制御回路18と同じ機能を提供する為のものであり、同じ例えばZ-80（商品名）等のCP

U等で構成されている。

【0070】以下、動作について説明する。

【0071】例えばネットワーク印刷を実行する場合の例を示すと、LAN 11からイーサネットトランシーバ9を介して入ってきたページ記述言語（例えば商品名CaPSL）は、DMAコントローラ10によって、直接メモリ6の特定領域（受信バッファ）に蓄えられる。そこで後述するページ記述言語インタープリタープログラムによってラスタイメージに変換される。JPEG圧縮画像は、ページ記述言語（以下、PDL）で伸長することもできるが、高速に伸長するためJPEG圧縮／伸長回路3が利用される。

【0072】なお、メモリ6は価格的な理由から、1ページの画面全体をラスタライズするだけの容量を持たず、PDL（例えばCaPSL）を部分的に処理するバンディングという手法を用いて、何回かに分けて処理される。ラスタライズが完了したら、第1のSPインタフェース回路4-1または第2のSPインタフェース回路4-2に転送され、各プリンターに出力される。なお、ネットワークスキャンを実行する場合の例は後述する。

【0073】図2は、図1に示したスキャナプリンタサーバーシステムを含むネットワークシステムの概略を示すシステムブロック図である。

【0074】本実施例におけるネットワークシステムは、例えばMacintosh（商品名）、IBM-PC（商品名）、SUN（商品名）の様な市販されているホストコンピュータのアプリケーションソフトウェアから、ネットワークを介して接続されている少数のスキャナやプリンターを共同利用する為の物である。同図に於て、大きく4つのブロックがあるが、左から順にMacintoshワークステーションST1、IBM-PCワークステーションST2、SUNワークステーションST3、及びカラーSPサーバーSP1である。MacintoshワークステーションST1、IBM-PCワークステーションST2、SUNワークステーションST3は、市販されているホストコンピュータであり、カラーSPサーバーSP1は発明者が設計したインタフェースユニットである。それらが、例えばEthernet（商品名）というローカルエリアネットワーク96によって相互接続されている。94は第1のカラースキャナ／プリンタ、95は第2のカラースキャナ／プリンタである。

【0075】MacintoshワークステーションST1、IBM-PCワークステーションST2、SUNワークステーションST3、及びカラーSPサーバーSP1は、相互通信の為に、共通の通信インタフェースを有する。51、61、71、81はローカルエリアネットワーク（LAN）96と通信する為のイーサネットトランシーバである。52、62、72、82はLAN96上に構築した通信プログラムTCP/IP（商品名）で、米国国防省の標準である。

【0076】このTCP/IPプログラムはEnd-to-Endでのデータ誤りの無い通信サービス（機能）を提供する。53、63、73、83は第1のカラースキャナ／プリンタ94、第2のカラースキャナ／プリンタ95をMacintosh用のステーションST1、IBM-PC用のステーションST2、SUNワークステーションST3から共同利用するという特別な目的のためのサービス（機能）を提供する通信プログラムである。53、63はS/Pクライアントプログラムとして、クライアント型のサービスを提供する通信プログラムであり、83はS/Pサーバープログラムで、サーバー型のサービスを提供する通信プログラムである。

【0077】MacintoshのステーションST1から第1のカラースキャナ／プリンタ94に印刷させる場合を説明すると、市販アプリケーションプログラム56で作成した原稿を印刷しようとする、プリントマネージャー（Printing Manager）55というOS（Operating System）の一部に制御が渡され、印刷のための制御を開始する。この時のデータフォーマットはMacintoshの場合QuickDraw（商品名）タイプに標準化されている。プリントマネージャー55は、変換プログラム54の描画関数群を原稿に書かれた通りにコールする。変換プログラム54は、そのコールの中でQuickDrawを、例えばCaPSL（Canon Printing System Language）コードに逐一変換して図示しないメモリーにCaPSLコードを蓄える。変換プログラム54は、主としてQuickDraw/CaPSL変換プログラムとして機能する。

【0078】通信プログラム53は、得られたCaPSLコードをTCP/IPプログラム52に従ってイーサネットトランシーバ51、LAN96、イーサネットトランシーバ81を介し、さらにTCP/IPプログラム82、通信プログラム83を介してカラーSPサーバーSP1に伝送する。なお、上述した変換プログラム54、64、74、通信プログラム53、63、73、スキャナインタフェースプログラム57、67、TCP/IPプログラム52、62、72等は、例えばフロッピーディスク等でホストコンピュータに供給されても良い。また、イーサネットトランシーバ、例えばボード回路としてホストコンピュータに供給されても良い。

【0079】カラーSPサーバー（S/Pサーバー装置）SP1全体の動作は、システム全体制御プログラム93によって制御されている。前記CaPSLコードの受信も、システム全体制御プログラム93に知らされる。システム全体制御プログラム93は、後述するPDLインタープリタープログラム84に対して、CaPSLコード化された原稿を、ビットマップ画像にラスタライズする様に要求する。ビットマップ画像にラスタライズされた画像データは、デバイスドライバ86に渡され、例えば第1のプリンタ制御ボード91を介してビデオインタフェースVideo I/Fから第1のカラースキャナ／プリンタ9

4に送られ、印刷される。

【0080】IBM-PCのステーションST2から印刷する場合も同様で、上記データの流れと同様であるが、IBM-PCのステーションST2の場合には、印刷原稿のデータフォーマットはGDI（商品名）に標準化されている。変換プログラム64は、主としてGDI/CaPSL変換プログラムとして機能する。

【0081】SUNワークステーションST3から印刷する場合は、上記変換プログラム54、64に相当する機能が標準化されていないため、アプリケーションプログラム75が直接通信プログラム73を介して通信する。ただし、市販のアプリケーションプログラム、例えばFrameMaker（商品名）から印刷できるようにする為、MIF/CaPSL変換用の変換プログラム74を間に入れている。

【0082】原稿をスキャン（読み取り）する場合は、例えばMacintoshのステーションST1の場合、スキャナアプリケーションプログラム58からスキャナインタフェースプログラム57を介して通信プログラム53にスキャンの要求を出す。すると通信プログラム53は、イーサネットトランシーバ51、LAN96、イーサネットトランシーバ81およびTCP/IPプログラム82を介し、通信プログラム83との間に、End-to-Endの通信路を確保し、スキャン命令を伝達するその命令はシステム全体制御プログラム93に伝えられ、スキャナ制御プログラムに対して第1のカラースキャナ/プリンタ94をスキャンさせるように要求する。

【0083】スキャンされた原稿の画像データは逆に、第1制御ボード91、デバイスドライバ86、通信プログラム83、TCP/IPプログラム82、イーサネットトランシーバ81、LAN96、イーサネットトランシーバ51、TCP/IPプログラム52、通信プログラム53、デバイスドライバ57、スキャナアプリケーションプログラム58の順に伝達される。

【0084】IBM-PCのステーションST2で原稿をスキャンする場合も同様であるが、SUNワークステーションST3からスキャンする場合は、デバイスドライバ57、スキャナインタフェースプログラム67に相当する機能が標準化されていないため、スキャナアプリケーションプログラム76が直接通信プログラム73を介して通信する。なお、スキャナインタフェースプログラム57、67はスキャナアプリケーションと通信プログラムとのデータの制御を行う。

【0085】カラーSPサーバーSP1には、もう1台のスキャナ/プリンタである第2のカラースキャナ/プリンタ95が接続されており、第2の制御ボード92を介して第2のカラースキャナ/プリンタ95を同様に制御する。

【0086】システム全体制御プログラム93を始めとするカラーSPサーバーSP1の全てのプログラムは、

例えばVxWorks（商品名）と言うシステムプログラム（OS）の管理下で動作する。

【0087】このサービスの為の特別なプログラム（例えば通信プログラム73、変換プログラム74等）を持たない一般のUNIXワークステーションからの要求を受け付ける為、カラーSPサーバーSP1には通信プログラム83と並行して、準業界標準のlpr/lpd通信プログラム90も載せている。

【0088】図3は本発明に係るスキャナプリンタサーバーシステムと各ホストコンピュータとのネットワーク構築状態を示す図である。

【0089】この図に示すように、各ホストコンピュータHOST1～Nは、Macintosh（商品名）、IBM-PC（商品名）、SUN（商品名）等のホストコンピュータで、それぞれのOSによりデータ処理が制御されている。これらのホストコンピュータHOST1～Nには、LAN96との通信を行うためのイーサネットインタフェースボード97、画像圧縮伸長ボード98、CPUボード99がそれぞれ設けられている。なお、画像圧縮伸長ボード98は、メモリ上でのソフトウェア処理により実現するものであっても良い。また、画像データを入出力する際に、圧縮伸長を用いない場合には、画像圧縮伸長ボード98が不要となる。さらに、本実施例ではLAN96としてイーサネットを使用しているが、ネットワークの方法は、Apple talk（商品名）、Token Ring（商品名）等を利用するネットワークであってもいいし、通信プログラムもTCP/IPプログラムに代えてOSI、IPX（商品名）等で構成されるシステムでも本発明を適用できることは言うまでもない。

【0090】図4は、図1に示したメインCPU回路1、イーサネット回路2、JPEG圧縮回路3より構成されるボード回路の詳細構成を説明する回路ブロック図である。

【0091】図において、101は例えばIDT79R3051（商品名）等で構成されるCPUで、ボード回路全体を制御する。なお、ボード回路にはリアルタイムOSが搭載されている。このOS上で起動されている図2に示した通信プログラム83、システム全体制御プログラム93、スキャナ制御プログラム85、PDLインタープリタープログラム84等をCPU101が起動し、マルチプロセスに動作を制御する。

【0092】102は主記憶メモリで、CPU101のワークメモリとして機能する。なお、本システムが電源投入されると、EPROM107内に記憶されている上記各プログラムやSCSIポート112に接続されたハードディスク等の補助記憶装置またはネットワーク上のホストコンピュータに記憶されているプログラムを本メモリ上にダウンロードして配置される。このため、各プロセスのプログラム自体は主記憶メモリ102上にあ

り、ここで動作する。

【0093】103はバンドメモリで、画像データをラスタ方法に数ライン分蓄積できるバンドメモリである。当該バンドメモリ103は、PDLデータをビットマップデータに展開するためのメモリである。PDLは通常1ページ分のビットマップデータを作成してから、プリンタエンジン側にデータを送るが、本システムでは、1ページを数バンドに分割してビットマップ展開を行う手法をとる。これは、PDLインタープリタープログラム84がPDLデータの並べ替え作業を行うことにより実現する。当該バンドメモリ103に展開されたビットマップデータは、プリンタ側に送られる。その後、次のバンド分の展開が行われて、再びプリンタ側に送られる。この繰り返しにより、1ページ分のプリントが行われる。

【0094】また、スキャナ側から画像データを読み込む時にも、一時的に本バンドメモリに蓄積される。CPU101あるいは本システムでは記載していないダイレクトメモリアクセス(DMA)のブロック転送機能を用いて、当該バンドメモリ103より画像データを読み出し、SCSIポート112に接続されたハードディスクあるいはイーサネットポート114に接続されたネットワークを介して、ホストコンピュータ側に転送される。1バンド分のデータが転送されると、再び次のバンド幅分のスキャンデータを入力して繰り返す。

【0095】104はメモリアクセスコントローラで、主記憶メモリ102、バンドメモリ103を構築するDRAMのアクセスとリフレッシュをコントロールする。通常DRAMは1ワード(8, 16, 32ビット等)毎アクセスする方法、ある一定長の連続アクセスする方法(ページREAD, Write)、DRAMのバンクを分けて交互にアクセスを繰り返し、アドレス発生は先行して発生させるインタリーブ方法のアクセス方法があるが、これらの方法を提供しメモリのアクセスの高速化を図っている。ただし、主記憶メモリ102、バンドメモリ103がSRAMで構成された場合には、リフレッシュ機能は必要はない。105はメモリクリアコントローラで、バンドメモリ103のデータを高速にクリアする。

【0096】図5は、図4に示したメモリクリアコントローラ105の詳細構成を説明するブロック図である。

【0097】図において、201はアドレス発生部で、クリアスタート信号に応じてアドレスバッファ部203にクリアすべきアドレスを出力する。202はデータラッチで、クリアデータをデータバッファ部204に出力する。

【0098】このように構成されたボード回路(メインCPU回路1、イーサネット回路2、JPEG圧縮回路3より構成される)において、まず、CPU101がバンドメモリ103内のデータを他の記憶装置あるいはイ

ンタフェースに転送されたことを確認すると、アドレスバッファ203、データバッファ204をイネーブルにする。それに対して、バンドメモリ103への他のアクセス手段をディセーブルにする。データラッチ202には、システムの立ち上げ時のイニシャライズであるデータ、例えば「00」を設定しておく。CPU101は、アドレス発生部201に対して、クリアスタート信号を送る。これに応じて、アドレス発生部201はアドレスを順次発生して、データラッチ部202に保持されたデータをメモリアクセスコントローラ104を通して、バンドメモリ103に書き込む。全メモリ分の書き込みが終了したら、アドレス発生部201は、CPU101にクリアエンド信号を送り、クリア動作を終了する。

【0099】クリア動作は、バンドメモリ103上の画像データがプリンタデバイスに転送されると、行われた次のバンドの画像データがPDLインタープリタープログラム84により展開される。その時、PDLインタープリタープログラム84は、必要な部分にしかビットマップ展開を行わない。例えば図6の(1)に示すバンドの後半の画像、(2)に示すバンドの展開は、(2)に属する画像のみが書込まれる。このため、図6の(1)で既に転送し終った画像データは、不要データとして残ってしまう。もし、メモリクリアをしなければ、(1)および(2)に示した画像が混在する画像データがバンドメモリ103上に書込まれた状態となってしまう。そこで、上記メモリクリアが必要となる。105はメモリクリアコントローラは、上記メモリクリア処理をハード化し、高速処理を可能としている。

【0100】画像描画処理回路106は、PDLインタープリタープログラム84の描画機能をハードウェアで補助するために構成された回路である。PDLインタープリタープログラム84は線を描いたり、図形の塗りつぶしが非常に多く、相当の時間を必要とする。例えば図7に示すような図形(1)~(3)に囲まれた部分の塗りつぶしは、図中矢印で示すように塗りつぶし処理が実行される。

【0101】図8は、図5に示した画像描画処理回路106の詳細構成を説明するブロック図である。

【0102】図において、106Aはアドレス発生部で、描画スタート信号に応じてアドレスバッファ部106Cにクリアすべきアドレスを出力する。106Bはデータラッチで、描画データをデータバッファ部106Dに出力する。

【0103】このように構成されたボード回路において、まず、CPU101がPDLインタープリタープログラム84による処理結果から、現在のバンド幅内に、線の描画や図形の塗りつぶしがある場合、画像描画処理回路106へ指令する。まず、データラッチ106Bに描画データをラッチさせる。次に、アドレス発生部106Aに開始アドレスとカウント数(1ライン分の描画量)を設定

する。ただし、カウント数は終了アドレスを設定する構成であっても良い。そして、アドレスバッファ106C、データバッファ106Dをイネーブルにする。それに対して、バンドメモリ103への他のアクセス手段をディセーブルにする。CPU101は、アドレス発生部106Aに対して、描画スタート信号を送る。これに応じて、アドレス発生部106Aは開始アドレス値より順次アドレスを発生して、データラッチ部106Bに保持されたデータをメモリコントローラ104を通して、バンドメモリ103に書き込む。カウント数部(終了アドレス)に達したら、メモリ書き込みを止めて、描画エンド信号を送り、次の開始アドレスとカウント数を設定し、再びメモリへ書き込みを行う。バンド幅分の描画を終了すると、バンド幅分の描画を終了する。

【0104】描画動作は、クリア動作と同様にハードウェアによる高速化を図ったものであが、例えば専用のLSIが描画処理をすべて受け持ち、CPU101による補助を軽減する構成としても良い。

【0105】一方、図4において、107は電源投入時に、CPU101が最初にアクセスするEPROMで、本システムをコントロールする上記各種プログラムが記憶されている。なお、EPROM107には、次の2つの記憶方法が可能である。第1の方法としては、プログラムがそのまま実行可能な形式で記憶する場合、第2の方法としては、プログラム全体を可逆的な圧縮方法により圧縮し、その形態で記憶しておき、電源投入後、伸長しながら、主記憶メモリ102に再配置する場合等である。

【0106】しかしながら、第1の方法による場合は、EPROM107は、そのまま実行可能なメモリとなるため、主記憶メモリ102に再配置されるより、主記憶メモリ102の節約になるが、EPROM107自身の容量を大きくすることと、アクセスが一般にEPROMは遅いという欠点がある。また、第2の方法による場合は、プログラム全体を圧縮するため、たくさんの量のプログラムをEPROM107に記憶させることができるため、EPROM107の記憶容量を削減できるとともに、主記憶上に再配置するため、アクセスが高速となる長所を有するが、主記憶メモリ102の容量を大きくする必要が生じる等の欠点もある。そこで、本実施例では第2の方法を採用している。

【0107】108はEEPROMで、電源が切られても記憶データを失わず、電源投入後の起動時に、データの書き換えが可能なデバイスである。このため、ネットワーク上のアドレスの記憶やスキャナ、プリンタ側の設定パラメータの記憶等に利用される。109は時計回路(RTC)で、時刻合せが可能であるとともに、電源OFF時でも、内部に持つバッテリーにより時間がそのまま進行する。これにより、実時間を知ることが可能となり、スキャン時やプリント時の情報として利用される。

110はタイマで、インターバルタイマである。本システムのプログラムはマルチプログラムで動作しているが、その管理は、リアルタイムOSが行っている。タイマ110は、数ミリsec毎のインターバルタイムをCPU101に通知しており、このタイマ110と優先順位スケジューラにより、各プログラムにCPU101を割り当てている。

【0108】111はSCSIコントローラで、周辺機器インタフェースの標準規格であるSCSIの制御を行う。112はSCSIポートで、周辺機器を接続するためのパラレル入出力I/Fポートである。113はイーサネットコントローラで、本システムをネットワークの1方式であるLAN96に接続し、データのやり取りを行うための制御部として機能する。イーサネットコントローラ113は、内部に小さなバッファメモリを有し、非同期式のCPUと同期式のネットワークとの時間的整合性を取っている。ネットワークより転送されてくるデータは、このバッファメモリに蓄えられ、逆にネットワークにデータを転送する場合もここから送信される。イーサネットコントローラ113は、電気的タイミングの制御とデータの転送、転入の制御を行っている。TCP/IPの通信プログラム82の制御は、イーサネットコントローラ113を用いてCPU101が制御している。

【0109】イーサネットポート114は太い同軸ケーブル114aを媒体として用いる。I/Fの形状は、15ピンのコネクタである。114bのTwist Pairタイプポートは、4線式の媒体を持ちいる。I/Fの形状は8ピンのモジュラジャックである。これらのポートを利用して、イーサネットワークに接続される。115はADCT圧縮/伸長回路は、RGB(RED, GREEN, Blue)で構成される中間調データ(各色8ビット)を、CCITT規格JPEG方式のアルゴリズムを用いた圧縮および伸長を行う回路である。スキャナより読み込まれたカラー中間調データを圧縮する場合は、スキャナ側から読み込まれた生画像データをADCT圧縮/伸長回路115を用いて圧縮し、小さなメモリ量にする。このためネットワーク転送のスピードアップや蓄積メモリの削減といった効果が得られる。また、ネットワークを介して転送されてきた圧縮画像データをADCT圧縮/伸長回路115を用いて伸長し、カラー中間調データに戻して、バンドメモリ103に書き込み、プリンタI/Fに転送されて、プリンタデバイスより印字される。

【0110】116はFIFOメモリで、ADCT圧縮/伸長回路115へ圧縮画像データを送ったり、受け取ったりするために用いられる。FIFOメモリ116は、圧縮動作、伸長動作時の圧縮データと伸長データ(生画像データ)のデータ量の違いによるデータの転送タイミングのずれを吸収するために用いられる。117

はRS232Cコントローラ部で、標準のシリアルI/Fの制御を行う。118はRS232Cポートで、Achannel118aとBchannel118bの2ポートが用意され、一方を端末接続用にして、ディスプレイ表示とキーボード入力を可能としている。また、もう一方はシリアルインタフェースを有するデバイス接続用になっている。ここには、レーザビームプリンタのようなキャラクタプリンタや簡易スキャナの接続が可能となっている。

【0111】なお、本システムでは、Achannelを端末接続用にしており、ここに端末を接続して、コマンドの入力やパラメータの変更を行うようにしている。ただし、本システムは疑似端末機能も有し、ネットワーク上のホストコンピュータより、rloginしてAchannelと同じ操作も可能としている。

【0112】119はセントロニクスI/Fコントローラ部で、セントロニクスI/F準拠のプリンタの接続のためのI/Fコントロールを行う。実際には、接続先のデバイスの状態(BUSY/non)を確認して、8ビットのデータを出力する。これの繰り返しでデータを転送する。その他セントロニクスI/Fに準拠した信号の制御を行っている。120はセントロニクスI/Fポートで、実際にケーブルが接続されて、プリンタとの電気的關係を結ぶ。121はVMEコントローラで、本システムのCPU101がVMEbusの規格に準拠した他のボードをアクセスするためのコントロールを行う。なお、VMEbus規格では、アドレスバスA16, A24, A32、データバスD8, D16, D24, D32のバスをアクセス可能に構成されている。バスの使用権は、アービトレーション方式より与えられる。その他、VMEbusの規格に適合した制御を本回路で実現する。122はVMEbusポートで、ダブルハイトのVMEbusを電気的に接続する。当該VMEbusポート122は、6ピンのコネクタ2つで構成され、この中にアドレスバス、データバス等が配置されている。

【0113】123はリセットスイッチで、本システムにおいて異常が発生した場合に、最終的手段として、システムを再立ち上げする場合に押下される。なお、リセットスイッチ123は、本実施例ではハード的なスイッチ機構であるが、上述したRS232Cポート118に接続された端末、あるいはネットワークからrloginした疑似端末から、プログラムでリセットするプログラムRESET(ウオームRESET)として構成しても良い。

【0114】124はLED部で、電源ON時を示すLED124a、CPU101が実行中を示すLED124b、各プログラムが自由に点灯可能なLED124c等より構成されている。これにより、LED124aが点灯中であれば、本システムが電源の入湯状態であることを操作者が視覚的に確認することができる。また、L

ED124bが点灯中であれば、CPU101が実行中であることを、LED124bが消灯中であれば、CPU101がウェイト状態であることを操作者が視覚的に確認することができる。頻繁に点灯するLED124cは、何かの処理が現在実行されていることを操作者が視覚的に確認することができ、点灯時間、間隔、回数等の相違により、どのプログラムによりLEDが点灯したか等を識別可能となる。

【0115】以下、ネットワーク上のホストコンピュータからのプリント要求処理におけるデータの流について詳述する。

【0116】ネットワーク、すなわちLAN96に接続されたホストコンピュータがデータをプリントしたい場合、SPサーバスシステムにデータと出力先を指定する情報を送る。この際、SCSIポートにハードディスクが接続されている場合、データは一旦ハードディスクに蓄積(スプール)される。また、ハードディスクがない場合は、主記憶メモリ102に一旦蓄積される。転送されてくるデータ形式としては、下記(1)～(3)の3通りである。

【0117】(1) PDL形式のデータ

(2) 生画像形成のデータ

(3) 圧縮画像形成のデータ

このうち、(1)のPDL形式のデータの場合には、PDLインタープリタープログラム84が起動され、バンド幅毎のビットマップ展開が行われて、バンドメモリ103に書き込まれる。バンドメモリ分のデータが揃うと、プリンタ側に転送され、次のバンド幅分のビットマップ展開が行われる。

【0118】一方、(2)の生画像形成のデータの場合は、バンドメモリ103にバンド幅分データが書き込まれ、指定されたプリンタ側に転送される。そして、次のバンド幅分のデータが用意される。生画像形式の場合、データ量が多いので、ホストコンピュータ側からは全部のデータを一括して転送するのではなく、分割して送る。そのため、プリントの最中もネットワークよりデータを受け取る作業が行われる。

【0119】さらに、(3)の画像圧縮形式のデータの場合は、圧縮画像データをFIFOメモリ116に書き込み、ADCT伸長回路115により生画像データに変換する。伸長後のデータは、バンドメモリ103にバンド幅分データが書き込まれ、指定されたプリンタ側に転送される。以下、同様の処理をそれぞれ繰り返す。なお、画像データの形式が、単一形式ではなく、上記データ形式が組み合わさった複合データの場合もある。すなわち上記(1)のデータ形式と(2)のデータ形式との組み合わせ、もしくは上記(1)のデータ形式と(3)のデータ形式との組み合わせの場合には、それぞれ上記の処理を行う。

【0120】画像データを転送するプリンタデバイスの

I/Fは、VMEバス122上に構成したインタフェースボードを介して接続する場合、RS232Cポート118に接続した場合、セントロエクスI/Fポート120に接続したことがある。これは、ホストコンピュータがどのI/Fに接続されたプリンタにより印刷したかにより行先が決定される。

【0121】バンドメモリ103上のデータは、1バンド幅分のデータ転送が終了すると、クリアコントローラ105がメモリクリアを行う。その後、再びPDL展開を行い、バンド幅が揃うと、指定されたプリンタ側のインタフェース回路にデータが転送され、デバイスドライバ86によるインタフェース回路(制御ボード)の制御によりプリンタでの出力処理が行われる。この繰り返しである。ネットワーク上のホストコンピュータがスキャナより画像データを入力したい場合、本システムは、スキャナ起動のための指定情報をホストコンピュータからもらい、画像入力を開始する。スキャナデバイスのインタフェースは、VMEバス122上に構成したI/Fボードを介して接続する場合、RS232Cポート118に接続する場合がある。この指定もホストコンピュータが指定する。

【0122】スキャナから入力される画像データは、バンドメモリ103にバンド幅分蓄積される。スキャナからの画像データを指定されたホストコンピュータに転送する場合、スキャナ制御プログラム85によって次の2通りの加工が行われる。

【0123】第1の加工としては、生画像形式のデータ、ただし、画像情報のタグを付加したものを含み、第2の加工としては、圧縮画像形式のデータである。

【0124】このうち、データ加工形式が生画像形式のデータの場合には、バンドメモリ103上のデータを、順次ネットワークを介して指定されたホストコンピュータに転送する方法と、一旦SCSIポート112に接続されたハードディスクに蓄積される場合がある。いずれも、ホストコンピュータよりの指定情報で選択される。また、画像情報を記録したタグをつける場合には、それを付加して転送する。

【0125】一方、データ加工形式が圧縮画像形式のデータの場合には、バンドメモリ103上のデータは、ADCT圧縮回路115に送られ、圧縮されて、FIFOメモリ116に圧縮後のデータが書き出される。そして、FIFOメモリ116より読み出してはホストコンピュータへ転送し、次のバンド幅分の処理を繰り返し行い、圧縮画像データを得る。ハードディスクを有する場合は、一時的に蓄積するという点が違うだけで、他は上述と同様である。

【0126】以下、図9に示す回路ブロック図を参照しながら図1に示した第1のSPインタフェース回路4-1の詳細構成について説明する。

【0127】図9は、図1に示した第1のSPインタ

フェース回路4-1の内部構成を説明するブロック図であり、図1と同一のものには同じ符号を付してある。

【0128】図において、1000はCPUで、ROM1002に格納された上記各種制御プログラムに基づいてボード回路の内部バス1001に接続される各デバイスを制御する。1003はRAMで、CPU1000のワークメモリ等として機能する。1004はネットワークコントローラで、LAN96に接続されるステーションとのアクセスを制御する。1005はバンドメモリで、バンド幅分のデータを記憶する。1006は標準圧縮伸長部で、画像データの圧縮伸長を行う。1007はSCSIコントローラで、ハードディスク1008等が接続されている。

【0129】211は前記第1のSPインタフェース回路4-1を総括的に制御するCPUで、ROM212に記憶された制御プログラム(デバイスドライバ86を補足するプログラム)に基づいてボード内部バスに接続される各デバイスを制御するとともに、各部の初期設定やスキャナ、プリンタとのコマンドのやり取りを行う。213はRAMで、CPU211のワークメモリ等として機能する。214はDPRAMで、第1のSPインタフェース回路4-1とボード回路とのコマンドのやり取りを行うRAMで、CPU211とVMEバス16を介してボード回路のCPU1000が共に独立してメモリアクセス可能に構成されている。215、216はバッファメモリで、CPU211とCPU1000との衝突を避けるために機能する。

【0130】218はリアルタイム圧縮伸長部で、多値画像データをリアルタイムで圧縮伸長するものである。ここで、リアルタイムとは、ビデオインタフェース220を通じて入力される画像データの速度、例えば約15MHz(32ビット)で画像データのやり取りをする速度と同時間で処理することを意味する。圧縮メモリ17は、リアルタイム圧縮伸長部218で圧縮したデータを格納する。219はラインバッファで、8ラインのラスタ方向のデータを保存するメモリとして機能する。なお、当該ラインバッファ219はリアルタイム圧縮伸長部218からは、8×8のマトリクスでADCT圧縮に準じたランダムアクセスが可能となっている。217はDMAコントローラで、圧縮メモリ17とリアルタイム圧縮伸長部218との間をCPU211を介さずにデータを移動させるためのものである。

【0131】ビデオインタフェース220aは、スキャナ94A、プリンタ94Bとのインタフェースである。220bは制御インタフェースである。

【0132】以下、ホストコンピュータからの指定情報に基づく第1～第4のモード処理について図9を参照しながら詳述する。なお、各モード処理は、ROM1002に格納された画像入出力装置制御プログラム85とROM212に格納されたデバイスドライバ85を実行す

ることによって行われる。

【第1のモード処理】先ず、第1のモード処理（画像プリントモード処理）では、メインCPU回路1のバンドメモリ1005内の生画像データを、VMEバス16を経由して受け取り、リアルタイム圧縮伸長部218で圧縮した後、圧縮メモリ17に一旦、圧縮画像の形態で蓄えられる。この動作は、1ページ分行われ、1ページ（1画面）分の圧縮が終了したところで、プリンタ94Bを起動する。次に、圧縮メモリ17に一旦、圧縮画像の形態で蓄えられた圧縮画像データをDMAコントローラ217が読み出し、リアルタイム圧縮伸長部218に送る。この際、伸長回路により、生画像データに復元する。伸長された生画像データは、順次ビデオインタフェース220よりプリンタ94Bへ転送される。なお、その際画像伸長動作はプリンタ94Bのデータ処理速度に同期して、超高速に伸長する。また場合によっては、画像処理を施した後にプリンタ94Bにデータを送り、プリンタ処理を行う。

【第2のモード処理】第2のモード処理（画像スキャンモード処理）では、スキャナ94Aより、ビデオインタフェース220aを介して入力されてくる画像データを、そのスピードに同期して超高速に、リアルタイム圧縮伸長部218の圧縮機能を用いて画像圧縮を行い、DMAコントローラ217により、出力される圧縮画像データを圧縮メモリ17に一旦格納する。1画面分のスキャンが終了したところで、ボード回路のCPU1000は再びリアルタイム圧縮伸長部218の伸長回路により圧縮メモリ17に蓄えられた圧縮画像データを伸長し、生画像データに戻す。その生画像データは、VMEバス16を経由してボード回路に送られる。ボード回路は、ホストコンピュータ側が、生画像データを要求している場合はそのまま指定されたホストコンピュータへ転送し、もし、圧縮画像データを要求した場合には、ボード回路上の圧縮機能処理により圧縮して指定されたホストコンピュータへ転送する。

【第3のモード処理】第3のモード処理（標準圧縮画像プリントモード処理）では、ボード回路が圧縮画像データをネットワーク上のホストコンピュータから受け取ると、そこで伸長せずに、圧縮画像データのまま指定された第1のSPインタフェース回路4-1に送る。第1のSPインタフェース回路4-1では、圧縮メモリ17に圧縮画像データを蓄え、その後指定されたプリンタ94Bを起動し、圧縮画像データをリアルタイム圧縮伸長部218の伸長機能により、プリンタ94Bのプリントスピードに同期して画像伸長を行いプリントする。

【第4のモード処理】第4のモード処理（標準圧縮画像スキャンモード処理）では、スキャナ94Aの走査速度に同期して、超高速にリアルタイム圧縮伸長部218の圧縮機能により画像圧縮を行い、圧縮メモリ17に一旦圧縮画像データを蓄える。1画面分のスキャンが終了し

たところで、ボード回路のCPU1000は、圧縮画像データをそのままVMEバス16を経由して取り込む。

【0133】なお、上記第1および第2のモード処理では、生画像データで第1のSPインタフェース回路4-1とインタフェースするため、第1のSPインタフェース回路4-1内では画像通信に必要とされる標準の圧縮伸長処理を行うことは必ずしも必要ではなく、高速化やその他の目的に応じた独自の圧縮伸長が行える。

【0134】また、上記第2、4のモード処理では、スキャナ94Aの読取りデータを直接圧縮をかける場合について記述したが、スキャナの読み取りデータに線密度変換や色空間変換等の画像処理を施した後に、リアルタイム圧縮伸長部218が画像圧縮をかけても良く、制限するものではない。また、画像処理機能は、画像圧縮伸長部とスキャナ94A、プリンタ94Bとの間に設ける構成であっても良いし、画像圧縮伸長部をスキャナ94A、プリンタ94B内に設ける構成であっても良い。

【0135】以下、図10に示すブロック図を参照しながらリアルタイム圧縮伸長部218の構成および動作について説明する。

【0136】図10は、図9に示したリアルタイム圧縮伸長部218の詳細構成の一例を示すブロック図であり、特に本実施例では圧縮伸長方式として、ADCT方式を採用している。

【0137】データ圧縮時は、生画像データは、先ず外部のラインバッファ219に蓄えられた後で、色空間変換部221に入力されて、RGBデータからY、Cr、Cb等の色空間に変換される。そして、場合によっては、Cr、Cbは色差成分として、サブサンプリングされ、画像の持つ冗長度を落される。次いで、8×8画素毎にDCT計算部222において、周波数空間に変換される。次に、図11に示すようにDCT係数はジグザグスキャン部223でスキャンされ、量子化部224で量子化される。この時に、量子化の係数は、量子化テーブル225に記憶されていて、DCT係数の8×8に対応するようになっている。次に、内部FIFOメモリ226に一旦蓄えられ、所望のタイミングでハフマン符号化部227でハフマンテーブル228を参照して符号化した後にでき上がった圧縮画像データは、外部のホストコンピュータ等からアクセスできるCODECレジスタ229に値が格納される。

【0138】ところで、色空間変換部221、DCT計算部222、ジグザグスキャン部223、量子化部224、量子化テーブル225等でパイプライン動作部が構成され、タイミング生成用のクロックに同期して動作し、高速で動作する。また、ハフマン符号化部227、ハフマンテーブル228、CODECレジスタ229等で非同期動作部が構成され、外部CPUがCODECレジスタ229をアクセスするスピードに同調して動作するように構成されており、CPUのスピードやDMAに

間に合う程度であって、あまり高速には動作しない部分である。また、パイプライン動作部は、スキャナ94A、プリンタ94Bの画像転送クロックに追従できるように高速動作が可能に構成されている。従って、内部FIFOメモリ226は、同期系のパイプライン動作部と非同期動作部の動作スピードの緩衝のために設けられている。

【0139】また、画質を高めるために圧縮率を下げるのと圧縮データが増え、非同期動作部の処理量が多くなり、非同期動作部の処理スピードが間に合わなくなる。ただし、色空間変換部221の生画像データのインタフェースは、スキャナ94A、プリンタ94B以外にも接続可能で、外部にFIFOメモリを設けて、CPU等からの非同期アクセスを受けることが可能で、この場合はパイプライン動作部を低速で動作させたり、一時停止させることが可能である。従って、非同期動作部はスピード的に問題はなくなる。

【0140】また、本実施例では画像圧縮伸長部を複数（例えば2つ）有し、生画像データを複数に分割して、それぞれの画像圧縮伸長部に与えることにより高速動作するスキャナやプリンタ等に接続することが可能となる。このように、複数に分割された画像データの圧縮であるならば、伸長時にも同様にスピード問題は解決される。

【0141】伸長時の動作は基本的には圧縮時の逆プロセスになるわけであるが、図10に示すように、圧縮画像データは、CODECレジスタ229に渡されると、ハフマン符号化部227でハフマンテーブル228を参照しながら逆ハフマン符号化、すなわちデコードされる。その値は一旦内部FIFOメモリ226でスピードの緩衝を受けながら、量子化部224で逆量子化される。逆量子化は量子化テーブル225の8×8の量子化係数を乗算することで行われる。そして、ジグザグスキャン部223で逆ジグザグスキャンされて、DCT計算部222にDCT係数として渡される。そして、色空間変換部221において圧縮色空間であるY、Cr、Cb等よりもとのRGB空間等に戻される。

【0142】また、DCT計算部222では、DCT計算、逆DCT計算は、計算のパラメータを入れ替えるだけで同一回路で処理される。色空間変換部221でも、その変換が線形一次変換等の場合には、同様にパラメータ変換で行える。

【0143】さらに、量子化部224も同様に量子化は割り算等、逆量子化は掛け算を行うが、割り算も、割る量子化係数を逆数で持つことにより掛け算になるので、同一回路で圧縮伸長とも可能である。以下、図12～図15を参照しながら第1のSPインタフェース回路4-1の各モードの動作を例にしてさらに詳細に説明する。

【0144】図12は、第1のSPインタフェース回路4-1のパート構成の概略を説明するブロック図であ

る。

【0145】図において、1100は画像処理制御部で、VMEバスインタフェース、画像圧縮伸長部、CPU回路部等から構成され、後述する図13に詳細を示す。1200はつながりメモリ制御部で、後述する図14に詳細を示す。1300は画像入出力制御部で、スキャナ、プリンタを備えるカラーレーザ複写装置（CLC）、スキャナ、プリンタを備えるバブルジェットカラー複写装置（BJC）とのI/Oを制御する。なお、詳細は図15に示す。

【0146】図13～図15は、図1に示した第1のSPインタフェース回路4-1の詳細内部構成を説明する回路ブロック図であり、図1と同一のものには同じ符号を付してある。

【0147】図9に示すボード回路のCPU1000より、画面を2分割されたデータのまず右面についての処理において、画像データはVMEインタフェース301よりまずFIFOメモリ302にデータがバッファ303を介して書き込まれる。FIFOメモリ302よりはバッファ304を介して一旦SRAM305に蓄積される。SRAM305の動作は、画像圧縮伸長部のパイプライン部に同期して動作する。RAM305に画像の垂直方向8ライン分のデータが蓄積されると、画像圧縮伸長部306は水平、垂直8×8単位ずつSRAM305より読み出して動作し、得られた圧縮データはDMAコントローラ部307を介してDRAM308に順次書き込まれる。この時、DMAコントローラ部307はアドレスカウンタ309を操作してアドレスを発生し、またはカウントアップしてセレクト310を介してDRAM308に与える。なお、第3のモードでは、VMEインタフェース301およびセレクト310を介してボード回路のCPU1000がアドレスを与え、かつインタフェース変換部312を介してDRAM308とのデータ更新を行うことになる。これにより、直接圧縮データをDRAM308に送ることが可能であり、画面を左右に2分割にしていない標準の圧縮データを扱うことも可能である。ただし、低い圧縮率のデータには不向きである。

【0148】また、DRAM308はリフレッシュ回路313より常にリフレッシュが行なわれ、データの維持がなされる。次に、DRAM308のデータをプリンタ94Bに出力する動作を説明すると、DRAM308よりDMAコントローラ307により書き込み時と同様な制御で圧縮伸長部306にデータが読み出されると、伸長が行なわれ、生データがバッファ304からFIFOメモリ302からの出力データはセレクトにおいて画像の左右のいずれかの面が選択されて、ラッチ315に与えられる。

【0149】次に、ガンマ変換部325でガンマ変換やLOG変換される。そして、マスキング回路316でC

G用のマスキングを行ない、マスキング回路317で、自然画用のマスキングを行ない、セクタ321で画像の性質に応じてどちらかを選択し、最終的に出力ガンマ調整部322で変換されてからラインドライバ324を介してプリンタ94Bに出力される。この時、出力ガンマ調整部322はガンマ設定レジスタ323に調整される。また、マスキング回路316はROMで構成され、マスキング回路317はロジックであって、黒生成のための黒テーブルが接続されるとともに、初期値データROM319の値が初期にロードされるようにタイミングコントローラ320も周辺に接続されている。プリンタ94Bが面順次の場合は、伸長動作を面の数だけ行ない、そのマスキング回路316、317は、出力色に応じたマスキング処理を行なう。

【0150】以上が、画面を2分割にした場合の右面の処理の流れであり、左面についても同様の処理の流れが行なわれる。すなわち、バッファ325、FIFOメモリ302、バッファ327を経由してSRAM328に蓄えられた8ラインの画像データは、画像伸長部329において圧縮され、DMAコントローラ331、アドレスコントローラ335、セクタ334によりコントロールされて、DRAM332に圧縮データとして書き込まれる。また、VMEインタフェース311より直接セクタ334をインタフェース変換部330を介してボード回路のCPU1000が圧縮データを書き込んでも良いことは言うまでもない。また、リフレッシュ回路333がDRAM332のデータをリフレッシュし維持する。そして、画像プリント時には、DRAM332より読み出されたデータがDMAコントローラ331等のコントロール331等のコントロールを受け、画像圧縮伸長部329で伸長され、SRAM328、バッファ327、FIFOメモリ326を経由してセクタ314に与えられ、以下上述の通り画像処理されて出力される。以上のようにして、モード1の処理が行なわれる。

【0151】次に第2のモード処理動作について説明する。

【0152】スキャナ94Aより各種の処理を受けて送出される画像データは、ラインシーバにより受け取られ、バッファ341により画像の右面がバッファ342により画像の左面が、FIFOメモリ303、326に順次取り込まれる。それ以降は、第1のモードと同様の動作を行なう。すなわち、右面の処理動作について記述すれば、FIFOメモリ302に書き込まれた画像データは、バッファ304を経由してSRAM305に8ライン分が読み出されて書き込まれる。次に、圧縮伸長部306によりDRAM308に圧縮データが生成される。

【0153】一方、第4のモードでは、VMEインタフェース311及びインタフェース変換部312、セクタ310を介してボード回路のCPU1000が直接圧

縮画像データを読み込むことが可能であるが、第2のモードでは、DRAM308の圧縮データを伸長してからボード回路のCPU1000に渡す。DRAM308よりインタフェース変換部312およびDMAコントローラ307、アドレスカウンタ309、セクタ310によりコントロールされて、書き込まれた順に読み出された圧縮画像データは、画像圧縮伸長部306で伸長され、SRAM305に8×8単位毎に書き込まれ、8ライン分すべてが書き込まれた時点でバッファ304を経由して、水平ライン方向に連続して読み出され、FIFOメモリ302に書き込まれる。FIFOメモリ302では、ボード回路のCPU1000のタイミングに従って、バッファ303、VMEインタフェース301を介して、ボード回路のCPU1000に生画像データとして読み出される。

【0154】ところで、セクタ321が2つのマスキング回路316、317の選択については、RAM351にCG画像があるか、自然画像であるかの属性が書かれていて、水平方向カウンタ352、垂直方向カウンタ353より、処理画素位置のアドレスが生成され、セクタ355を介してRAM351に与えられ、その読み出しデータにより、セクタ321を制御する。従って、RAM351上の属性はボード回路のCPU1000よりVMEインタフェース301、セクタ351を介し、RAM351上にあらかじめ書き込まれる。また、スキャナ94A、プリンタ94Bとのインタフェースは、S/Pタイミングコントローラ356により制御されている。

【0155】次に、画像の左右面の切り替えの制御について説明する。

【0156】垂直方向カウンタ363は、画像の垂直方向の有効部分について画像有効信号を送出するもので、ディレイレジスタ364は、画像先端の余白部分をセットするものである。長さレジスタ361は画像の有効長をセットし、比較器362において有効長以内で有れば、各部分に画像有効信号を送るものである。垂直方向の画像有効区間に加えて、水平方向の余白長をセットするディレイレジスタ366が接続され、画像の左端をカウントした後に有効信号を発する。カウンタコントローラ部367は、これを受けて水平左幅カウンタ368を起動する。この水平左幅カウンタ368は、ダウンカウンタとして機能し、幅分だけカウントを行ない、その間に左画面のイネーブル信号を発する。左画面が終了すると、カウンタコントローラ部372に対してイネーブルを発し、カウンタコントローラ部372は右画面の幅カウンタ373で幅分のカウントを行ない、終了状態を制御部389に与える。制御部389はセクタ314に対して画像有効幅の終了を知らせて出力をクリアする。幅カウンタ368は、圧縮時にはライトパルスジェネレータ369を左画面有効幅で発生してマルチプレクサ3

71を介してFIFOメモリ326に左面のスキャナデータをかき込み、FIFOメモリ326はゲート381を介して圧縮伸長部329のリードパルスマルチプレクサ371を介してFIFOメモリ326に与える。

【0157】また、伸長時には、幅カウンタ368の左面有効信号をリードパルスジェネレータ370が受けて、マルチプレクサ371を介してFIFOメモリ326よりプリンタ94Bへ出力させる。

【0158】一方、画像圧縮伸長部329のライトパルスがゲート381を介してマルチプレクサ371を介してFIFOメモリ326に与えられ、伸長データが書き込まれる。

【0159】右面についても同様にライトパルスジェネレータ374、リードパルスジェネレータ375、マルチプレクサ376、ゲート382が同様の動作をすることは言うまでもない。また、画像圧縮伸長部329には内部のFIFOメモリ302の状態がスタート/ストップロジック386に与えられるとともに、周辺状態を検出するレジスタ385もスタート/ストップロジック386に与えられ、かつ外部のFIFOメモリ326の状態もFIFOメモリコントローラ部383を介してスタート/ストップロジック386に与えられ、内部のFIFOメモリ302及び外部のFIFOメモリ326がオーバーフローしたり、アンダーフローしたりしないように管理され、状態に応じて画像圧縮伸長部329のパイプライン部329をストップさせたり、スタートさせたりすることができる。

【0160】同様に、状態レジスタ387、スタート/ストップロジック386、フィホコントローラ部384に管理され、圧縮伸長部306のパイプライン部がスタートしたり、ストップしたりする。なお、スタート/ストップでは、圧縮時に内部のパイプライン部が速すぎて内部のFIFOメモリ302がオーバーフローしそうな時や、伸長時に内部のパイプライン部が速すぎて内部のFIFOメモリ302がアンダーフローしそうな時の他の圧縮時に画像圧縮伸長部が速すぎて外部FIFOメモリ326がアンダーフローしそうな時、および伸長時に画像圧縮伸長部306が速すぎて外部FIFOメモリ326がオーバーフローしそうな時がある。

【0161】また、エラー対策399は、スキャナ94A、プリンタ94Bが速すぎて画像圧縮伸長部306の処理速度が間に合わない時に、外部のFIFOメモリ326や内部のFIFOメモリ302がオーバーフローやアンダーフローを起こすので、スキャナ94A、プリンタ94B側のビデオ信号(ビデオ=画像)を一時的に止めるもので、スキャナ94Aでは1ライン単位でビデオが捨てられ、プリンタ94Bでは1ライン単位で余白が出力されるように簡易的なエラー処理が行なわれ、FIFOメモリをオーバーフローまたはアンダーフローされてビデオの流れを破壊することを防ぐ。従って、エラー

対策部は、エラー情報を制御部388に渡すと、制御部389が1ラインの余白、左面有効部、右面有効部を把握して、1ラインの区切り目でエラー解除をエラー対策部399に出力する一方、エラー時にスキャナ94A、プリンタ94Bの入出力データをクリアするように各部に指示するとともに、エラー処理中に画像圧縮伸長部306がスキャナ94A、プリンタ94Bの処理速度に間に合わなかったにもかかわらず、1水平ラインの区切りまで圧縮または伸長動作をさせる。このことにより、1ラインの区切りで再びエラー復帰ができるのである。ただし、エラー処理中に圧縮では、白ラインを圧縮して圧縮速度をかせぎ、伸長では画像データはプリンタ94Bに間に合わないために捨てられる。

【0162】以上の説明の中では、ボード回路のCPU1000がほとんどすべての管理を行なうように説明したが、本実施例では第1のSPインタフェース回路4-1の中にも内部のCPU392を有していて、第1のSPインタフェース回路4-1ないで処理できることは内部のCPU392が処理を分担することが可能で有る。また、第1のSPインタフェース回路4-1内にはデュアルポートRAM393を有し、両ポートはそれぞれVEMバスインタフェース311を介してボード回路のCPU1000が内部CPUバスを介して内部のCPU392に接続されている。このデュアルポートRAM393を介して、ボード回路のCPU1000と内部のCPU392に情報伝達が行なわれている。

【0163】従って、第1のSPインタフェース回路4-1内のVMEバスへの2つのCPU1000およびCPU392からのアクセスは衝突することはない。この時、CPU392の内部バスに接続されているバス変換部394により画像圧縮伸長部306、329とのインタフェースがなされる。バス変換部394は、インタフェース変換部312、330と同様の機能を有し、レジスタインタフェース395を介してボード内の各種のレジスタとのインタフェースを行なうので、VEMバスのインタフェースとしての機能をも合わせて有している。

【0164】さらに、内部のCPUバスには、CPU392のプログラムを格納するROM391、シリアル通信部390を有している。シリアル通信部390はスキャナ94A、プリンタ94Bとの動作制御をするコマンドが送られる。従って、本実施例では図9に示すボード回路のCPU1000は図13に示す内部のCPU392を介してスキャナ94A、プリンタ94Bを制御するか、ボード回路のCPU1000が直接スキャナ94A、プリンタ94Bを制御する構成となっている。なお、コントロールのコマンドとしては、実行コマンドとしてスキャナ94A、プリンタ94Bの起動命令の他に、状態検知コマンド、状態設定コマンドがある。状態検知コマンドは、プリンタ94Bの紙有り/紙無し、カセット有り/無しサイズ、その他のトナ残量、ジャム発

生等多くのコマンドとともに、現在設定されている動作モード、すなわち単色／3色／4色や解像度、その他の画像処理パラメータの検知等がある。スキャナ94Aに関しても、ほぼ同様で、その他の原稿台上の原稿有り／無しや、ランプ切れ等を知ることが可能である。

【0165】一方、状態設定コマンドは、プリンタ94Bではカセットサイズ選択、カセット上下段選択、動作モードやプリント枚数設定や画像処理系のパラメータ設定等である。

【0166】なお、スキャナ94Aについても同様で、例えば変倍率設定、2値／多値化設定、標準色空間変換、独自色空間変換、線密度（解像度変換）、領域指定設定、ガンマ変換設定等がある。

【0167】なお、本実施例ではスキャナ94Aに多くの画像処理機能を持つために、第1のSPインタフェース回路4-1上では画像処理は行わずに圧縮する。また、スキャナ94AよりRGBデータとしてデータ受信する。

【0168】一方、プリンタ94Bは入力CMYK入力であり、プリンタ部に画像処理をあまり有していないため、マスキング、ガンマ変換、LOG変換、CMYK生成は第1のSPインタフェース回路4-1上で処理される構成となっている。

【0169】図16は、図1に示した第2のSPインタフェース回路4-2の詳細構成を説明するブロック図であり、図2に示したバブルジェット型のスキャナプリンタ95（プリンタ95B、スキャナ95A）とS/Pサーバ装置SP1とのデータ処理をインタフェースする。また、第2のSPインタフェース回路4-2は一体としてインタフェースボードとして構成されている。

【0170】図において、401はCPUで、デュアルポートRAM403を介してVEMバスにつながっているボード回路からのコマンドを受け取り、解釈し、第2のSPインタフェース回路4-2内部の制御をする。CPU401は内蔵されたインターバルタイマによって2ms毎の割り込みを発生させ、バブルジェット型のプリンタ95B、スキャナ95Aとのコマンドの通信を行う。また、画像処理回路部404、405、406の各種パラメータの初期化および変更を行う。

【0171】402はプログラム用のROMで、CPU401が実行する制御プログラム（デバイスドライバ86を補足するプログラム等）を格納するとともに、画像処理回路部404、405、406の初期値やプリセット値を格納する。

【0172】デュアルポートRAM403はCPU401のワークエリア等として機能するとともに、VEMバスによってつながれたボード回路のCPU1000と、CPU401の両方からアクセスことで、両者間の通信を行う。

【0173】画像処理回路部404は画像処理用ASICと

して構成され、ルックアップテーブルにより階調変換を行う。例えばRGBデータをCMYKデータに変換するときは、LOG変換を行う。これはあらかじめROM402に変換テーブルを用意しておき、これを画像処理回路部404内のRAMに転送することで実現する。

【0174】画像処理回路部405は画像処理用ASICとして構成され、4×5のマトリックス演算とルックアップテーブルにより階調変換を行う。このマトリックス演算によりスキャナ95Aのセンサの特性のRGB空間と標準色空間としてのNTSC-RGB色空間の変換または画像処理回路部404の変換した後のCMY

(K)からプリンタ95Bの特性に合わせたCMYKへの変換（マスキング処理と呼ばれる）などを行う。さらにルックアップテーブルによってカラーバランスの調整が行える。これらの処理も画像処理回路部404と同様にあらかじめ様々なテーブルを用意しておき、ROMに格納しておき、用途に合わせて、選択して設定する。

【0175】画像処理回路部406は画像処理用ASICとして構成され、データの2値化処理を行う。なお、本実施例に置ける二値化のアルゴリズムは平均濃度保存法であるを採用している。407はシリアル／パラレル変換部でありスキャナ95A、プリンタ95Bへの通信のためにCPU401からの8ビットのパラレルのデータをシリアルデータに変換する。

【0176】408は画像用DRAMである。プリンタ95のヘッドの画素数に合わせたバンド形式に合わせた大きさの画像用メモリである。スキャナ95Aまたはプリンタ95Bでは1回の走査の間は画像用クロックに従ってデータが流れ、動作を止められないために、1バンドの大きさでのバッファリングが必要である。このためスキャナ95Aによってスキャンされるまたはプリンタ95Bにプリントされる1バンド分の画像データをバッファリングを行う。

【0177】また、VMEバス側からのラスタ形式のアクセスとスキャナ95A、プリンタ95Bよりの縦方向のアクセスとの走査形式の変換を行う。

【0178】409はDRAMへのアドレスセクタ、マルチプレクサで、DRAM408へのアクセスはVEMバス側からとスキャナ95A、プリンタ95B側からの2つがあり、これらからのアドレスの切り替えを行う。さらにDRAM408へのアドレスはROWアドレスとCOLUMNアドレスに分けて供給するのでこのマルチプレックスを行う。

【0179】410はDRAMのタイミングコントローラ部である。RAS、CAS、WE、OEなどのDRAMを制御する信号を作り出す。またリフレッシュ信号との調停を行う。

【0180】411はタイミング回路で、スキャナ95A、プリンタ95Bのアクセスのタイミングを生成する。これはスキャナ95A、プリンタ95Bからの画像

クロックや同期信号をもとにアクセスのタイミングを作り出す部分である。

【0181】412はリフレッシュタイミング制御部である。DRAM408へのリフレッシュのタイミングを作り出す。スキャナ95A、プリンタ95Bのアクセスのすきまを使いスキャナ95A、プリンタ95Bのアクセスとが衝突しないように制御している。

【0182】413はVMEタイミング制御部である。VMEバスからのアクセスのための制御信号の処理をする。AMコードのデコードや上位アドレスのデコード、割り込みの処理などである。

【0183】414はスキャナ95A、プリンタ95Bのアクセスアドレス生成部である。スキャナ95A、プリンタ95Bのアクセスは通常のラスタ形式とは異なっているため、VMEバスからのアクセスのためにラスタ形式でかかれたメモリに対して、スキャナ95A、プリンタ95B用の特殊なアクセスのアドレスを生成する部分である。これはバンドの大きさで、走査方向の縦横をひっくり返すものである。

【0184】415はVMEバスインタフェースデータバッファ部である。画像データは32ビット幅、コマンドは8ビット幅でアクセスをする。

【0185】416はVMEバスインタフェースアドレスバッファ部である。画像データは24ビットのアドレス空間、コマンドは16ビットのアドレス空間でアクセスする。

【0186】417はVMEバスインタフェースのデータとアドレス以外の部分のバッファである。

【0187】418は画像処理部の入り口のバッファである。VMEバスからのアクセスおよびDRAMへアクセスは32ビット幅で行うが、画像処理部では8ビット幅で処理がなされる。このため32ビットのR、G、B、Xのデータを8ビットのデータとしてR、G、B、Xの順にシリアルに変換をする。

【0188】419は画像処理部の出口のバッファで、バッファ418とは反対に、色順次に変化する8ビット幅のデータラインを4色分まとめて32ビットにする変換を行う。

【0189】420は2値化処理後のバッファで、画像処理回路部406によって2値化されたデータは1ビットになっている。これを8ビットに拡張する。すなわち、「0」は「0x00」に、「1」は「0xFF」にする。

【0190】421はバッファで、画像処理回路部406による2値化処理をバイパスする場合用のバッファで、バッファ420とこのバッファ421の出力のどちらか一方を選択して、2値と多値を切り替える。

【0191】422はスキャナ95A、プリンタ95Bのインタフェースの通信部のバッファである。

【0192】423はスキャナ95A、プリンタ95B

のインタフェースの入力データ用バッファである。

【0193】424はスキャナ95A、プリンタ95Bのインタフェースの出力データ用バッファである。

【0194】425はスキャナ95A、プリンタ95Bのインタフェースのクロック、制御信号用の入力バッファである。

【0195】426はスキャナ95A、プリンタ95Bのインタフェースのクロック、制御信号用の出力バッファである。429はクロック入力ラインである。

【0196】430は32ビットの画像用データバス、431は24ビットのアドレスバス、432は8ビットの画像データバス、433は8ビットの画像データバス、434は16ビットのローカルアドレスバス、435は8ビットのローカルデータバスである。以下、プリンタ95Bのプリント動作について説明する。

<プリント時の動作>まず、VMEバスを通してボード回路より、プリントするにあたっての各種のパラメータがデュアルポートRAM403に書き込まれると、CPU401はこのデータを読み出して解釈して制御を行う。例えば、RGBデータのプリントであれば、CPU401は画像処理回路部404のLUTデータの変化を起こさないスルーの特性のテーブルをセットし、画像処理回路部405のマトリックスの係数テーブルにはNTSC-RGBからBJ-RGBへの変換用の係数をセットし、画像処理回路部406の2値化処理をスルーするように、バッファ420、421のゲートを制御する。

【0197】さらに、データのサイズ等のパラメータをセットする。そしてデータサイズなどのパラメータはの平行/シリアル変換部407を通してプリンタ95Bへ伝えられる。次に、1バンド分の画像データがVMEバスを通して、ボード回路よりメモリ408に転送される。このときVMEバスからは32ビットアクセスでRGBXのデータ形式でストアされる。RはRED、GはGREEN、BはBLUEのそれぞれの色成分の画像データであり、Xは黒文字用の情報を含んだ制御用データである。次にデュアルポートRAM403を介してプリント動作のコマンドが伝えられる。CPU401はプリント動作の開始命令をプリンタ95Bへ伝える。プリンタ95Bのプリンタ制御部から開始信号が帰ってくるとタイミング発生部411メモリ408へのアクセスを始める。このときデータの読み出しはプリンタ95BのBJヘッドに沿った方向であるため、アドレス発生部414によって生成されたアドレスに従って読み出しを行う。メモリ408から読み出されたデータはバッファ418でR、G、B、Xの順に8ビットデータに変換され画像処理部に入る。あらかじめ設定したパラメータによって、画像を処理し、NTSC-RGBデータはプリンタ95Bの内部で使われるRGB色空間に変換され、バッファ421、インタフェース424を通してプリンタ95Bへ伝えられる。1バンド分のデータの処理が終わ

ったら、次のバンドのデータをVMEバスを介して受け取り、上記動作を繰り返す。所定の回数の処理を終えたら1ページの処理が終了する。以下、ROM402に格納された制御プログラムによるスキャナ95Aの原稿読み取り動作について説明する。

＜スキャン時の動作＞まず、VMEバスを通してボード回路より、スキャンするにあたっての各種のパラメータがデュアルポートRAM403に書き込まれる。CPU401はこのデータを読み取って解釈して、制御を行う。例えば、RGBの2値データで、1024×1024の大きさで512×512の位置からのスキャンとすると、CPU401は画像処理回路部404のLUTにスルーの特性のテーブルをセットし、画像処理回路部406のマトリックスの係数テーブルにはBJR-RGBからNTSC-RGBへの変換用の係数をセットし、画像処理回路部406の2値化処理を通してバッファ420、421のゲートを制御する。さらに、スキャンする画像のサイズを1024×1024に、スキャンの開始位置を512×512に設定する。これらのパラメータはパラレル/シリアル変換部407を通してスキャナ95Aへ伝えられる。次にCPU401はスキャナ95Aへスキャンの開始のコマンドを伝える。スキャナ95Aのスキャナ読取り部より入力された画像データはインタフェース423を通して画像処理回路部404、405、406に入力される。ここで予め設定したパラメータによる画像処理を行い、バッファ419によってRGBXの32ビットの形式のデータとしてメモリ408にストアされる。この時、メモリ408にはRGBXのデータが入っているが、この例での設定ではRGBの2値画像をスキャンするため、Xは意味のないデータであり、R、G、Bの各成分は2値データであるが1画素1バイトである。これを一般的な2値画像の要求する形式に、例えば8画素1バイトのパッキングをしラスタライン順次にRGBを配置するといった処理はボード回路において行う。メモリ408にある画像データはVMEバスインタフェース415を介してボード回路へ転送される。上記の処理をバンドの回数分繰り返して、1回のスキャンの動作を終了する。

【0198】以下、図1に示したスキャナ94A、プリンタ94Bの画像データ処理について説明する。

【0199】本実施例では図1に示したスキャナ94A、プリンタ94Bと一体としてカラー画像複写装置が形成されているため、画像処理機能は1系統しか有しておらず、スキャナ94A、プリンタ94Bを分離すると、1つの処理につき、どちらか一方しか有していない構成となっている。しかも、大部分の画像処理機能はスキャナ94Aに備えられ、プリンタ94Bは主として変倍、領域指定、色空間変換、ガンマ変換、色マスキング処理部等がある。

【0200】スキャナ94Aには、色空間変換部や色マ

スキング処理部等を有しているので、ビデオインタフェースの切り口として、標準RGB(NTSCのRGB等)が用意されていて、RGBのデータは点順次またはパラレル同時に得られる。従って、プリンタ94Bに対しては、画像処理部を有していないので、C、M、Y、Kでデータを与えなければならず、外部でC、M、Yへの変換や黒(K)生成、色マスキング処理、その他必要に応じて解像度変換、トリミング等の画像処理を行ってから、ビデオインタフェースに送らなければならない。しかもその場合は、面順次に各色ずつC、M、Y、K4回繰り返し、画像を送出を行わなければならない。また、スキャナ/プリンタともに途中で動作を停止したり、中断するようなことはできない。

【0201】その他ビデオインタフェースには、水平同期信号、垂直同期信号およびビデオクロックが含まれてビデオデータとの同期をとっている。また、スキャナやプリンタの電源ON/OFF等のステータス情報も有しているので外部からの確認が可能である。また、コマンドインタフェースをシリアル通信で行なう機能も有しており、これにより、スキャナ94Aやプリンタ94Bの状態検知や状態設定、スキャナ94Aやプリンタ94Bの起動等の実行コマンド等を発行することができる。

【0202】以下、図17及び図18を参照しながらスキャナ95A、プリンタ95Bの動作について更に説明する。

【0203】図17は、図2に示したスキャナ/プリンタ95の画像記録プロセスを説明する模式図である。

【0204】図において、101Rはスキャンする原稿を表し、102Rはプリントする用紙を表す。これらは、例えばA4サイズである。103Rはスキャナのセンサのヘッドを表し、104Rはプリンタのヘッドを表す。プリンタのヘッド104Rはバブルジェット方式によってインクを吹き出すノズルが並んでおり、例えば128のノズルから構成されている。

【0205】一方、センサは128より多い画素をスキャンできるように、例えば144画素のデータを出力できる。これらのヘッドはカラーの場合、スキャナのセンサではRGBの3色分、プリンタヘッドではCMYKの4色分が並んで構成されている。105Rは画像処理部であり、スキャナセンサより入力したRGB信号を処理し、プリンタヘッドに合わせた特性のCMYK2値信号として送る。

【0206】画像処理部105Rはその画像処理系の途中のインタフェース部106RからRGB各色8ビットのデータを外部とやり取りすることができる。スキャナ/プリンタ95Bではスキャナ95Aのセンサとプリンタ95Bのヘッドが同期して動き、画像処理部105Rはパイプライン構成となっているため、大きな容量の画像メモリを持たずに処理がなされている。このためインタフェース105Rでやり取りされるデータの走査形式

は特殊なものとなっている。107Rは外部機器である。

【0207】図18は、図2に示したスキャナ／プリンタ95Bのスキャナ95Aの原稿走査状態を示す模式図である。

【0208】図において、201Rはヘッドの動きを示している。ヘッド自体は原稿（用紙）に対して図のように横方向（主走査方向）に動く、センサの各画素はこれとは垂直に配列している。このため、データ203Rが並ぶ。一方、一般的なラスタ走査形式ではデータ204Rのように並ぶ。

【0209】図19は、図2に示したスキャナ／プリンタ95Bのスキャナ95Aのバンド原稿走査状態を示す模式図である。

【0210】図において、301Rは1ページを示し、302Rは第1のセグメントを示し、303Rは第2のセグメントを示している。スキャナセンサから出力され画像処理系を通り2値化されるまでの画像では、セグメント304Rのようにセグメント305Rより大きな画像が扱われ、幅306Rの大きさだけ重複して処理がなされる。

【0211】以下、図20～図22を参照しながら本実施例に示したS／Pサーバー装置が制御可能なプリンタの一例について説明する。

【0212】本実施例に示したS／Pサーバー装置が制御可能なプリンタとしては、セントロニクス・インターフェースを利用可能なプリンタである。セントロニクス・インターフェースは、米国セントロニクス社が自社のプリンタ用に開発したコンピュータからプリンタにデータを送るための規格で、安価でかつ高速のデータを送ることができる。現在のプリンタは、ほとんどこのセントロニクスが標準となっている。

【0213】セントロニクスのデータ伝送は、図20に示すようにDATASTROBE信号、ACKNOWLEDGE（ACK）信号、BUSY信号用の3本の制御線とDATA線によって行なう。

【0214】ここで、DATASTROBE信号は、DATA線に、データが出力されたことを示す。BUSY信号は、現在プリンタが動作中であり、データを受け取れないことを示し、または、データ・バッファがフルであることを示す。

【0215】ACK信号は、データの読み取りが正常に終了したことを示す。

【0216】基本的には、上記3本の制御線で十分であるが、プリンタの制御を考えて、紙切れなどの信号線も定義してある。図20には、信号名、入出力、備考を示した。ピン番号は36ピン、25ピン、14ピンとコネクタの種類がまちまちな事と、各社で若干定義が変わっていたり、削除されている場合も多いため割愛した。

【0217】図2.1は、セントロニクスI／F制御回路

の一例を示す回路ブロック図である。

【0218】図において、201AはセントロニクスI／F制御回路で、データバッファ202A、制御線バッファ203Aを備え、図22に示すタイミングチャートに従ってデータ処理が行なわれる。

【0219】図23は、図21に示したセントロニクスI／F制御回路によるホスト－プリンタ間の信号処理手順の一例を示すフローチャートである。なお、(1)～(3)は各ステップを示し、特にホスト側の処理に対応する。

【0220】まず、BUSY信号が「L」で、かつACK信号が「H」となったら(1)、データを設定し(2)、DATASTROBE信号を出力して(3)、ステップ(1)に戻る。

【0221】図24は、図21に示したセントロニクスI／F制御回路によるホスト－プリンタ間の信号処理手順の一例を示すフローチャートである。なお、(1)～(6)は各ステップを示し、特にセントロニクス対応プリンタ側の処理に対応する。

【0222】まず、BUSY信号が「L」となり(1)、BUSY信号を「H」とし(2)、データをデータバスより取り込みを開始する(3)。次いで、データ取り込みを終了し(4)、ACK信号が「L」とし(5)、BUSY信号を「L」、かつACK信号を「H」に設定し(6)、ステップ(1)に戻る。データ転送はこのようにして行なわれる。

【0223】セントロニクス・プリンタは上記転送条件のもと、“ESC”（0x1B）をコマンドやデータの先頭に付けた方法で制御される場合が多い。例えば、あるプリンタにデータを転送する場合、ESC（A COUNT COLORDATA,（1B 28 4 1 COUNT COLOR DATA）等のように送る。

【0224】ここで、「ESC（A）」は制御用のコードである。「COUNT」はデータ数である。「COLOR」はRGB、CMYなどの色空間定義である。

「DATA」はカラー画像データである。このようなデータコマンドを連続してプリンタに送れば、プリントが行なわれる。なお、制御コードや構成は各社のプリンタによって違っている。しかしながら、“ESC”を用いた制御方法は、比較的類似している。

【0225】本システムは、セントロニクス・ポートにセントロニクス・プリンタを接続し、制御コードを、ソフトウェアプログラムでサポートすることにより、各種セントロニクス・プリンタを利用することができる。

【0226】図25は本発明に係るスキャナプリンタサーバー（ネットワークサーバー）SP1とホストコンピュータとのプログラム構成を説明する図である。以下、システム全体の流れを概略的に説明する。なお、図2と同一のものには同一の符号を付してある。また、図25

でのホストコンピュータは図2に示すMacintoshのステーションST1を例にして説明するが、他のステーションST2、ST3等であっても構わない。

【0227】ホストコンピュータ（ステーション）ST1のオペレータがアプリケーションプログラム56を用いて作成した印刷データをプリントするため、所望とするスキナプリンタネットワークサーバー、プリンタ、紙サイズ、送出するデータ形式等を選択指示すると、アプリケーションプログラム56は変換プログラム54にデータ（指示情報を含む）を通信する。変換プログラム54はアプリケーションプログラム56から送られたデータを選択されたネットワークサーバーSP1が受け付けるデータ構造に変換し、通信プログラム53、TCP/IPプログラム52に通信する。例えばMacintoshのステーションST1では、QuickDrawデータからCaPSLデータに変換し、IBMPCのステーションST2では、GDIデータからCaPSLデータに変換する。

【0228】通信プログラム53は、変換プログラム54から送出されたデータをTCP/IPプログラムを介してネットワークサーバーSP1に通信し、ネットワークサーバーSP1の通信プログラム83がTCP/IPプログラム82を介して受信し、システム全体制御プログラム93に通信する。システム全体制御プログラム93は、送られてきたデータを解析し、その時のネットワークサーバーSP1の状態と送られてきたデータに従い、以下の処理を行う。

【0229】システム全体制御プログラム93は、印刷データをPDLインタープリタープログラム84に送る。PDLインタープリタープログラム84は印刷データを受け取り、指定されたプリンタ（例えばスキナプリンタ95のプリンタ）が受け入れ可能なデータに変換する。例えばポストスクリプト（商品名）やCaPSLというPDLのデータから画像データに変換する。システム全体制御プログラム93はPDLインタープリタープログラム84が変換したデータを画像入出力装置制御プログラムとしてのデバイスドライバ86に送り、デバイスドライバ86はデータを指定されたプリンタに送りプリントさせる。

【0230】次に、ホストコンピュータST1のオペレータがスキナアプリケーション58を用いて画像入力するため、所望とするスキナプリンタネットワークサーバー、スキナ、画像の領域、解像度、カラーあるいはモノカラー、圧縮の種類等を選択指示すると、スキナアプリケーション58はスキナインタフェースプログラム57を介して通信プログラム53に通信する。通信プログラム53は、スキナインタフェースプログラム57を介して送られたデータをTCP/IPプログラムを介して指定されたネットワークサーバーSP1に通信し、ネットワークサーバーSP1に通信し、ネットワ

ークサーバーSP1の通信プログラム83がTCP/IPプログラム82を介して受信し、システム全体制御プログラム93に通信する。システム全体制御プログラム93は、入力した選択指示命令（画像入力命令）をスキナ制御プログラム85に送り、スキナ制御プログラム85は、命令に従って、画像入出力装置制御プログラムとしてのデバイスドライバ86に命令を送り、デバイスドライバは指定されたスキナ（例えばスキナプリンタ95のスキナ）を起動して画像データを入力し、画像データをスキナ制御プログラム85へ送り、スキナ制御プログラム85は画像データをシステム全体制御プログラム93へ送り、システム全体制御プログラム93は通信プログラム83へ送り、ネットワークサーバーSP1の通信プログラム83がTCP/IPプログラム82を介して指定されたホストコンピュータの通信プログラム（例えば通信プログラム53）へ画像データを送信する。通信プログラム53がTCP/IPプログラム52を介して受信した画像データは、さらにスキナアプリケーションプログラム58に送られる。

【0231】次に、ワークステーションST1から指定されたスキナプリンタサーバーへ（例えばスキナプリンタサーバーSP1へ）スキナプリンタサーバーの状態を問い合わせる命令が送られた場合は、システム全体制御プログラム93がネットワークサーバーSP1の状態（例えばネットワークサーバーSP1に接続されているスキナプリンタの種類、解像度、紙サイズ、色処理能力等）を取得し、TCP/IPプログラム82を介して、通信プログラム83が指定されたホストコンピュータ（例えばワークステーションST1）に通信する。

【0232】また、画像入出力装置としてのスキナプリンタ94、95またはネットワークサーバーSP1内でエラーが発生した場合は、システム全体制御プログラム93がエラーの状況を管理し、通信プログラム83がTCP/IPプログラム82を介して指定されたホストコンピュータ（例えばホストコンピュータSP1）に通信する。

【0233】以上のように、本実施例ではこれらのプログラム構成により、ホストコンピュータのアプリケーションプログラム（例えばDTPソフト）からホストコンピュータで指定したプリンタでプリントすることができる。また、ホストコンピュータのスキナアプリケーションプログラム（例えばDTPソフト）からホストコンピュータで指定したスキナから画像を入力できるし、指定したスキナで入力した画像を別のホストコンピュータへ送出することもできる。また、指定したネットワークサーバーSP1の状態（接続されるスキナプリンタの状態）を確認することができる。

【0234】なお、図25では、LAN96に接続されるホストコンピュータ、スキナプリンタサーバーはいくつ接続されていても本発明の適用を妨げるものではな

い。以下、ホストコンピュータ、ホストコンピュータとネットワークサーバSP1との間のネットワーク処理について説明する。

【0235】ホストコンピュータにおいて、プリントプロセスを行なう場合、大きく分けて第1～第3の処理、すなわち第1はアプリケーションプログラム（例えばDTPソフト）によるデータの作成処理、第2はアプリケーションプログラムによって作成されたデータの上記CaPSLコードへの変換処理、第3はCaPSLコードのネットワークサーバSP1への転送処理である。

【0236】なお、第1のデータの作成処理で作成されたデータは、使用するマシンの機種、アプリケーションプログラムに依存する。例えばSUNワークステーションでFrame Technology社製のDTPプログラムであるFrameMaker（商品名）を使用した場合、MIF（商品名）ファイルまたはIPL（商品名）ファイルで出力される。また、使用するマシンの機種がIBM社製のIBM-PCでWindows（商品名）対応のアプリケーションプログラムを使用した場合、GDI関数の呼び出しになる。また、アップル社製のMacintosh（商品名）を使用した場合、QuickDraw関数の呼び出しになる。

【0237】また、アプリケーションプログラムによって作成されたデータの上記CaPSLコードへの変換処理では、第1の処理で作成されたデータをCaPSLコードに変換するので、マシンの機種、作成されるファイル形式に依存する。例えばSUNワークステーションで上記Frame Makerを使用した場合、MIFファイルまたはIPLファイルをCaPSLコードに変換するプログラムとする。また、使用するマシンの機種がIBM社製のIBM-PCでWindows対応のアプリケーションプログラムを使用した場合、GDI関数からCaPSLコードへの変換を変換プログラムで行なう。さらに、アップル社製のMacintosh（商品名）を使用した場合、QuickDraw関数をCaPSLコードへの変換を変換プログラムで行なう。

【0238】さらに、第3はCaPSLコードのネットワークサーバSP1への転送処理においては、送信するデータはCaPSLであるが、送信処理は送信する際に使用するプログラムに依存する。例えば図2に示す通信プログラム83を使用したり、UNIXをOSとしているホストコンピュータではlpq/lpdプログラム90を使用する。

【0239】また、ホストコンピュータにおいてスキャナプロセスを行なう場合、大きく第1、第2の処理が行なわれる。第1にイメージデータをネットワークサーバSP1から受信する。第2にスキャナアプリケーションプログラムでイメージデータの表示、保存を行なう。

【0240】第1の処理において、受け取るイメージ

データは、第2の処理で使用するスキャナアプリケーションプログラムが扱えるイメージデータ形式に保存する。また、ネットワークサーバSP1からイメージデータを受信する際に使用するプログラムとしては、図2に示す通信プログラム83を使用する。

【0241】第2の処理において、スキャナアプリケーションプログラムの使用に依存して、扱えるイメージデータ形式が決まる。入力、出力各々について扱えるイメージデータ形式を定める。例えばビットマップおよびTIFF形式を入力として許可する。また、出力としてビットマップおよびTIFF形式を許可する。ファイルで保存する場合、TIFF形式を用いる。表示する場合にはビットマップを用いることができる。

【0242】ホストコンピュータとネットワークサーバSP1間をネットワークで接続する時は、複数のプログラムを使用する。中心となるプログラムは図2に示す通信プログラム83である。

【0243】ネットワークサーバSP1では、例えば図25に示す通信プログラム53とネットワークサーバSP1側の通信プログラム83がある。ホストコンピュータ側の通信プログラムは、主として第1～第3の処理を行なう。

【0244】第1は下位レイヤを介して、ネットワークサーバSP1とリンクを張る。第2はCaPSLデータをネットワークサーバSP1へ送信する。第3はネットワークサーバSP1からイメージデータを受信して上位レイヤへ送る。

【0245】一方、ネットワークサーバSP1は、主として第1、第2の処理を行なう。第1はTCP/IPプログラム82を介して通信プログラム83で受信したCaPSLデータをシステム全体制御プログラム93へ送る。第2はシステム全体制御プログラム93から受け取ったイメージデータを通信プログラム83へ送信する。

【0246】プリント処理及びスキャン処理は、ホストコンピュータの上位レイヤからのプリントおよびスキャンの要求を通信プログラム53が受け取ることで開始される。通信プログラム53は、下位レイヤを使用して、リンクを張る。例えばイーサネットを介して接続した場合、下位レイヤとして、TCP/IPプログラム52を使用してデータの送受信を行なう。通信プログラム53では、リンクを張った後に、プリントおよびスキャンに特有の情報の受け渡しをして、適合する条件にプリンタ及びスキャナを設定する。プリンタおよびスキャナの設定が終了後は、プリントはCaPSLをクライアントからサーバへ、またスキャナの設定が終了した後は、プリント時はCaPSLをクライアントからサーバへ、またスキャン時はイメージデータをサーバからクライアントへ各々送信する。プログラムとして、図2に示したlpdプログラム90も使用できる。ホストとして、U

NIXマシンを使用した場合に、UNIXの標準のプリントコマンドであるlprを使用してプリントが可能となる。このlpdプログラム90(図2参照)はプリント時のみに使用する。サーバー側にスプールディスクがない場合でもデフォルトの設定での出力は可能である。

【0247】システム全体制御プログラム93の基本的な動作は、イベントを入力してイベントに応じた処理を行なうことと、エラー中の入出力装置に状態を問い合わせ、エラーから回復していた場合、エラーによって中断していたジョブがあれば再開することである。

【0248】以下、図26に示すフローチャートを参照しながら本発明に係るスキャナプリンタサーバーシステムにおける全体制御動作について説明する。

【0249】図26本発明に係るスキャナプリンタサーバーシステムにおける全体制御手順の一例を示すフローチャートである。なお、(1)～(6)は各ステップを示す。

【0250】まず、ステップ(1)でイベントがあるかどうかを判別し(1)、ある場合はステップ(2)でイベントに応じた処理を行ない、ステップ(1)でイベントが無い場合は、ステップ(3)でエラー中の入出力装置があるか調べ、ある場合はステップ(4)でエラー中か問い合わせ、ステップ(5)でエラー中か判別し、エラーから回復していればステップ(6)でエラーにより中断していたジョブがあれば再開する。

【0251】システム全体制御プログラム93はネットワークサーバーSP1内でTCP/IPプログラム82、通信プログラム83、PDLインタープリタープログラム84、スキャナ制御プログラム85、デバイスドライバ86と通信することができ、さまざまなイベントを受けとる。通信プログラム83から送られるイベントとして、ジョブの到着、データ転送の終了があり、PDLインタープリタープログラム84から送られるイベントとして、受信データ処理終了、画像データ描画終了、ページ描画終了、プリントジョブ終了があり、スキャナ制御プログラム85からのイベントとして、画像データ転送要求、画像入力処理終了、ページプリント終了、エラー発生、正常状態等のイベントがある。

【0252】また、システム全体制御プログラム93では各イベントに対する動作が定まっており、例えば、通信プログラム83からジョブの到着が送られると、ジョブの内容を解析し、プリントジョブの場合はPDLインタープリタープログラム84にデータを転送し、画像入力ジョブの場合はスキャナ制御プログラム85にデータを転送する。

【0253】さらに、デバイスドライバ86から送られるエラーとしては、紙なし、インク切れ、紙詰まり、電源OFF等がある。

【0254】なお、本実施例では、ネットワークサーバーSP1では、プリンタのページ記述言語としてCaP

SLを採用している。そこで、CaPSLの機能処理等について説明する。なお、プリンタのページ記述言語としては、CaPSLに限定されるものではない。

【0255】CaPSLの機能は、図形、文字、イメージを表現している制御コードを入力して、ネットワークサーバーSP1のバンドメモリと呼ばれる描画領域に画像を展開するタスクである。

【0256】PDLインタープリタープログラム84と直接コミュニケーションを持つプログラムは、システム全体制御プログラム93である。コミュニケーションは、以下の第1、第2のコミュニケーションである。

【0257】第1のコミュニケーションは、システム全体制御プログラム93がCaPSLに引き渡すコミュニケーションであり、その内容は、CaPSLコードの格納されたファイル名、CaPSLコードが格納されてあるメモリの先頭アドレスおよびサイズ、バンドメモリ描画可能性等である。

【0258】第2のコミュニケーションは、PDLインタープリタープログラム84がシステム全体制御プログラム93に引き渡すコミュニケーションであり、その内容は、バンドメモリ描画終了、ページ描画終了、ドキュメント描画終了、バンドメモリ描画領域、空バンドメモリの情報である。

【0259】この際、PDLインタープリタープログラム84の内部では、以下の処理が行われる。

【0260】ネットワークサーバーSP1では、バンドメモリと呼ばれる、ある幅を持ったメモリを持っているので、PDLインタープリタープログラム84は画像をバンド幅分だけ部分的に展開しなければならない。PDLインタープリタープログラム84は、図27に示すようなプリンタ制御コマンドを入力する。CaPSLコードは、PDLインタープリタープログラム84内部でレイアウトと呼ばれるプログラム(図示しない)に基づいて図28に示す中間コードに置き換えられる。本実施例において、中間コードとは、いわゆる抽象的な図形の表現を具体的なデバイスの表現にしたものである。そして、1ページ分の中間コードができあがると、ペインタと呼ばれるプログラム(図示しない)に基づいて中間コードを参照しながらバンドメモリにラスティメージを展開する。

【0261】例えば図29に示すような図形を描くようなCaPSLコードをレイアウトが受け取ると、デバイスに依存した図28に示す中間コードに置き換える。ここで、レイアウトは、図形の大きさに基づいてその図形が何バンド目から何バンド目まで描かれているかを計算して、中間コードに登録する。図29では、円(circle)が1～2バンド、矩形(Rectangle)が2～4バンドに描かれる場合を示す。なお、その後に書かれている数字は、デバイス座標系における、図形の位置や半径等である。

【0262】次に、ペインタ（実際にメモリに描画するプログラム）は中間コードを参照しながら、図29に示すような図形を展開して、バンドメモリに描画する。実際には、メモリはバンド幅分の大きさしか持っていないので、バンドを描画すると、そのデータをプリンタに送り、メモリをクリアして、次のバンドのデータを描き始める。第0番目の描画をするために、中間コードを参照すると、第0番目のバンドに描くべきデータがないことが分かり、次のバンドに移る。すると、ペインタは第1バンド目に描くデータをサーチし、円を検知して1バンド分だけ描画する。次に第2バンド目に移り、円と矩形を描画することが分かるので、円の続きの部分の描画と矩形を1バンド分だけ描画する。このようにして、ペインタは、1ページ分のデータを描画するのである。

【0263】なお、CaPSLが展開するデータの1画素の構造は、図30に示されるように、RGBXの32ビットになっている。これは、色のデータを表わすRGB24ビットと、付加情報のための8ビットの付加情報Xから構成されている。ここで、付加情報Xの中は、ビット0がイメージ領域判定ビットであり、CaPSLがイメージをバンドメモリに描画する時には、このビットをONにする。また、ビット2は、黒情報ビットになっており、CaPSLで展開された図形や文字の色がRGBともに0の場合は、このビットをONにする。これらの情報は、第1のインタフェース回路4-1でプリンタ94Bでプリントアウトする際に解析され、画質を向上させる情報となる。

【0264】以下、図31に示すデータ処理経路図を参照しながら、スキャナ94A、95Aの動作について説明する。なお、図25と同一のものには同じ符号を付してある。

【0265】図31は、図2に示したスキャナ94A、95Aの動作を説明するデータ処理経路図である。なお、画像をバンド単位で切り分けてスキャンするスキャナの場合と、1度に全画像をスキャンする場合のタスクの動作が多少異なるためそれぞれを説明する。

【0266】画像をバンド単位で切り分けてスキャンするスキャナの場合において、先ず、クライアントマシンとしてのホストコンピュータST1から送られてきた、スキャン命令C1をTCP/IPプログラム82を介して通信プログラム83が受信すると、システム全体制御プログラム93にスキャン命令C2の到着を知らせる。

【0267】システム全体制御プログラム93は、この装置がスプール1を持つ場合は、受信したスキャン命令C2を元にしてスプールファイルを作って保存し、そのファイル名C3をスキャナ制御プログラム85に送る。スプール1を持たない場合は、スキャナコマンドC3を直接スキャナ制御プログラム85に送る。

【0268】スキャナ制御プログラム85は、スキャナコマンドC3を解釈し、解像度等のスキャナ条件設定命

令C4をデバイスドライバ86に与えて指定されたスキャナを起動する。デバイスドライバ86は、それぞれのスキャナ制御プログラム85に対応して存在する関数である。

【0269】スキャナ条件を設定したデバイスドライバ86は、リターン値C5をスキャナ制御プログラム85に返す。これを受けるとスキャナ制御プログラム85は、スキャン開始命令C6をデバイスドライバ86に与えて指定されたスキャナを起動する。デバイスドライバ86は、スキャナEを作動させて画像データを読み込みバンドメモリFに書き込む。1バンド分書き込むと、スキャナ制御プログラム85にスキャン終了のリターン値C9を返す。これを受け取ったスキャナ制御プログラム85は、バンドメモリFの画像データに画像処理、例えばJPEG圧縮ボードを用いることで圧縮を行ってバッファ領域Hへ書き込む。

【0270】バンドメモリF内のデータを全て処理し終わると、スキャナ制御プログラム85は、システム全体制御プログラム93に転送要求C11を送り、また同時にデバイスドライバ86にスキャン開始命令C6を与えて起動する。このスキャンは、前回のスキャンの時の終了点から読み込むようにスキャナ制御プログラム85で制御する。

【0271】一方、転送要求C11を受けたシステム全体制御プログラム93は、通信プログラム83に転送要求C12を送る。これを受け取った通信プログラム83は、指定されたクライアントマシンとしてのホストコンピュータにバッファ領域H内の処理された画像データを送る。このとき通信プログラム83、システム全体制御プログラム93は、スキャナ制御プログラム85とは、異なったプログラムであるので、スキャナ制御プログラム85内の関数であるデバイスドライバ86の制御で動くスキャナ作動中であっても、画像データの転送を行うことができる。

【0272】通信プログラム83は、転送が終了するとシステム全体制御プログラム93に転送終了信号C14を送り、これを受け取るとシステム全体制御プログラム93は、スキャナ制御プログラム85に転送終了C15を送る。スキャナ制御プログラム85は、デバイスドライバ86からのスキャン終了を伝えるリターン値C9、転送終了C15の両方を受けると、バンドメモリF内の画像データを画像処理しバッファ領域Hに書き込む。以上の処理を繰り返すことで画像データをホストコンピュータST1へ送ることができる。なお、別のホストコンピュータ（例えばホストコンピュータST2またはホストコンピュータST3）を指定することで、他のホストコンピュータへ画像データを送ることもできる。

【0273】全ての画像データの転送をスキャナ制御プログラム85が確認すると、システム全体制御プログラム93を介して通信プログラム83まで終了を知らせ

る。

【0274】一方、1度に全画像をスキャンする場合は、クライアントマシンとしての、例えばホストコンピュータST1から送られてきた、スキャン命令C1を通信プログラム83が受信すると、システム全体制御プログラム93にスキャン命令C2の到着を知らせる。

【0275】システム全体制御プログラム93は、この装置がスプール1を持つ場合は、受信したスキャン命令C2を元にしてスプールファイルを作って保存し、そのファイル名C3をスキャナ制御プログラム85に送る。スプール1を持たない場合は、スキャナコマンドC3を直接スキャナ制御プログラム85に送る。

【0276】スキャナ制御プログラム85は、スキャナコマンドC3を解釈し、解像度等のスキャナ条件設定C4をデバイスドライバ86に与えて指定されたスキャナを起動する。スキャナ条件を設定したデバイスドライバ86は、リターン値C5をスキャナ制御プログラム85に返す。これを受けるとスキャナ制御プログラム85は、スキャン開始命令C6をデバイスドライバ86に起動する。ここでのスキャナEは、読み込み動作を途中で止められないものとしている。

【0277】デバイスドライバ86は、命令C10によりスキャナEを作動させて画像データすべてを取り込み、取り込んだ画像データに画像処理、例えばJPEGボードGを用いてJPEG圧縮を行なってバッファHに描き込むことができる。

【0278】また、画像データを生のままバッファHに書き込むことも可能である。バッファHが一杯になると、スキャナ制御プログラム85は、システム全体制御プログラム93に転送要求C11を送る。これを受けたシステム全体制御プログラム93は、通信プログラム83に転送命令C12を出し、通信プログラム83がバッファHに描かれたデータを指定されたホストコンピュータST1へ転送する。なお、別のホストコンピュータ（例えばホストコンピュータST2またはホストコンピュータST3）を指定することで、他のホストコンピュータへ画像データを送ることができる。

【0279】通信プログラム83は、転送が終了するとシステム全体制御プログラム93に転送終了信号C14を送り、これを受け取るとシステム全体制御プログラム93は、スキャナ制御プログラム85に転送終了C15を送る。スキャナ制御プログラム85は、転送終了C15が送られてくるとバンドメモリ内の画像データを画像処理しバッファHに書き込む。以上の処理を繰り返すことで画像データをクライアントマシンとしての指定されたホストコンピュータへ送ることができる。

【0280】スキャナ制御プログラム85は、全ての画像データの転送を確認すると、システム全体制御プログラム93を介して通信プログラム83まで終了を知らせる。

【0281】以下、図32および図33を参照しながら本発明に係るスキャナプリンタサーバシステムにおけるホストコンピュータからS/Pサーバ装置へのデータの流れについて説明する。

【0282】ホストコンピュータから指定されたS/Pサーバ装置（ネットワークサーバ）SP1へのデータの流れにおいて、先ず、例えばホストコンピュータST1でDTPのアプリケーションプログラム56が実行されていてドキュメントを作成する。このドキュメントのデータ形式は、ホストコンピュータの機種、アプリケーションプログラムに依存したものである。そこで指定されたS/Pサーバ装置SP1が解釈できる形式のデータに変換する必要がある。その変換プログラムが変換プログラム54である。DTPのアプリケーションプログラム（DTPアプリ）56からはホストコンピュータのシステムに依存したデータが変換プログラム54に送られる。変換プログラム54は、受けとったファイルを、例えばCaPSLコードに変換する。さらに、CaPSLコードに変換したファイルは通信プログラムであるIpd505または通信プログラム53に送られる。

【0283】通信プログラムはホストコンピュータとS/Pサーバ装置SP1の間でネットワークを介して接続、通信を行なうためのものであり、ここでは2つの内のどちらか一方のプログラムを用意する。

【0284】先ず始めに、通信プログラム53を用いた時を説明する。

【0285】通信プログラム53に対応するプログラムとして、S/Pサーバ装置SP1側で通信を司るプログラムは通信プログラム83であり、このプログラム間ではTCP/IPプログラムを使用する。プリント時の通信プログラム53の役割は主に2つあり、一つは下位レイヤ（TCP/IP）を介して通信プログラム83とリンクを張り、通信できる状態にすることである。もう一つは、変換プログラム54が生成したCaPSLデータファイルをS/Pサーバ装置SP1に送信することである。

【0286】また、通信プログラム83の役割はリンクを張りプリント処理に必要な情報をやりとりして最適にプリントできるようにすること。通信プログラム53の送ってくるデータを受けとり、システム全体制御プログラム93にデータの到着を知らせる。この時、送られてきたCaPSLデータ、およびそれに付随する情報は受信バッファに一時的に貯められている。また、S/Pサーバ装置SP1側で何らかのエラーが起こった時に、その情報を通信プログラム53に送信するというような役割もある。

【0287】以下、通信プログラム83からプリントアウトまでの処理について説明する。

【0288】ここまでで、DTPアプリケーションプログラム56で作成したドキュメントデータはS/Pサーバ

バー装置SP1に送られたことになる。次に通信プログラム83はシステム全体制御プログラム93にジョブ（例えばカラーレーザ複写装置（CLC）からプリントアウト）の到着のイベントを送る。システム全体制御プログラム93はイベント駆動型のプログラムで通信プログラム83、デバイスドライバ86、PDLインタープリタープログラム84から送られてくるイベントを常に待っている。イベントが入ってくるとイベントの発信元と内容を調べてそれに対応した処理を行なう。今、システム全体制御プログラム93には通信プログラム83から「ジョブが到着した」というイベントが入力されている。この時、システム全体制御プログラム93はイベントを解析してハードディスク519を持っている時には受信バッファのドキュメントデータを一旦スプールする。そして、ジョブが幾つかたまっている場合には、ジョブの内容、プリンタ、スキャナの状態、ジョブの優先順位などを考慮して最適にジョブを起動する。

【0289】ハードディスク519がない場合には、データを貯めておくことが出来ないで、データの格納されているアドレスとサイズをPDLインタープリタープログラム84に知らせてプリントアウト処理を直ちに行なうように要求を出す。ここでは、ハードディスクがあるものとして説明を続ける。システム全体制御プログラム93はジョブの内容を判断してPDLインタープリタープログラム84にスプールファイル名を渡し、起動の要求をする。

【0290】以下、図33に示すブロック図を参照しながらさらに詳述する。

【0291】図33は本発明に係るS/Pサーバー装置SP1と、例えばカラーレーザ複写装置（CLC）とのデータ処理状態を説明するブロック図である。

【0292】図32に示したPDLインタープリタープログラム84はスプールファイルからCaPSLデータを読み込み、解釈して、図形、文字、イメージをバンドメモリ518に描画する。CaPSLデータの中に圧縮符合化されたイメージデータがあった場合にはPDLインタープリタープログラム84は、標準圧縮伸長部606でイメージデータを伸長してバンドメモリに描画する。1バンド分の描画を終了すると、PDLインタープリタープログラム84はシステム全体制御プログラム93に対して「1バンド描画終了」のイベントを発行する。イベントを受けとったシステム全体制御プログラム93はデバイスドライバ86に「1バンド排紙」の要求イベントを発行する。

【0293】以下、バンドメモリ605から圧縮メモリ615までのデータの流れについて説明する。なお、ここではデバイスドライバ86の動作について説明する。

【0294】「1バンド排紙」のイベントを受けとったデバイスドライバ86は、バンドメモリ605に格納されている展開データを、サブボード610Aのラインバ

ッファ617に転送する。ラインバッファ617に格納された展開データはブロック順次でリアルタイム圧縮伸長部616に転送される。リアルタイム圧縮伸長部616では、ブロック順次に得られる画像データを圧縮していき、順次圧縮メモリ615に格納する。圧縮メモリ615のアドレス生成や、ライト信号の生成はDMAコントローラ621が行なう。

【0295】1バンド分の圧縮が終了するとデバイスドライバ86はシステム全体制御プログラム93に対して「1バンド排紙終了」のイベントを送信する。イベントを受けとったシステム全体制御プログラム93は、PDLインタープリタープログラム84に対して「1バンド描画要求」イベントを出力する。このようにバンド毎に描画しては、圧縮するという処理を繰り返し1ページ分の圧縮展開イメージを圧縮メモリ615に格納する。最後のバンドの格納が終了すると、PDLインタープリタープログラム84はシステム全体制御プログラム93に対して「1ページ描画終了」のイベントを発行する。イベントを受けとったシステム全体制御プログラム93は、デバイスドライバ86に対して「1ページ排紙命令」を発行する。

【0296】「1ページ排紙命令」を受けとったデバイスドライバは、圧縮メモリ615に格納されている1ページ分の展開イメージをプリントアウトするために次のような処理を行なう。

【0297】CPU611はDPRAM622をONにしてからリアルタイム圧縮伸長部616、DMAコントローラ621の初期化および指定されたプリンタ620とのコマンドの通信を行ないリアルタイム圧縮伸長部616に伸長開始の命令を出す。リアルタイム圧縮伸長部616はDMAコントローラ621にアクセスし、DMAコントローラ621はアドレス、リード信号の生成を行ない、圧縮メモリ615より圧縮データがリアルタイム圧縮伸長部616に入力され、伸長された後ブロック順次でラインバッファ617に出力される。そしてラインバッファ617にてブロック順次からラスタ順次の変換をしてSP1/F618を通じて転送されプリンタ620に出力される。実際に、1ページ分のプリントアウトが終了するとデバイスドライバ86から「1ページ排紙終了」のイベントがシステム全体制御プログラム93に対して送られる。するとシステム全体制御プログラム93は、PDLインタープリタープログラム84に対して2ページ目の描画命令」を発行する。このようにして複数ページのドキュメントの出力が処理されて行く。

【0298】最終ページの最終バンドの描画が終了すると、PDLインタープリタープログラム84は「ドキュメント終了」イベントをシステム全体制御プログラム93に対して発行する。システム全体制御プログラム93はデバイスドライバ86に対して「1ページ排紙命令」を出しデバイスドライバ86はプリント処理を行ない

「1 ページ排紙終了」イベントをシステム全体制御プログラム 93 に対し発行する。なお、プリント終了は、システム全体制御プログラム 93 は必要ならば通信プログラム 83 に「プリント終了」イベントを発行する。通信プログラム 83 はホストコンピュータ側の通信プログラム 53 に対しプリント終了を知らせる。

【0299】また、エラー発生時には、例えばプリンタ 620 が、紙詰まりを起こしたり、紙無しの状態になった時にはデバイスドライバ 86 からシステム全体制御プログラム 93 に対して「エラー発生」のイベントが送られる。システム全体制御プログラム 93 はその旨を PDL インタープリタープログラム 84 と通信プログラム 83 に伝える。PDL インタープリタープログラム 84 はプログラムの状態の退避などのエラー時の処理を行ない、通信プログラム 83 はエラーの発生や、その内容をホストコンピュータ側の通信プログラム 83 に伝える。なお、エラーが回復したかどうかを調べる方法は 2 つ考えられる。一つは、ある一定期間毎にシステム全体制御プログラム 93 がデバイスドライバ 86 に対して問い合わせ、デバイスドライバ 86 が答えるという方法。

【0300】もう一つは、システム全体制御プログラム 93 がエラー回復を監視していて回復時にシステム全体制御プログラム 93 に対して「エラー回復」のイベントを発行するというものである。

【0301】以下、図 32、図 34 を参照しながらホストコンピュータから、例えばバブルジェットカラー複写装置へのプリントデータ出力処理について詳述する。

【0302】図 34 は本発明に係る S/P サーバー装置 SP1 と、例えばバブルジェットカラー複写装置 (BJC) とのデータ処理状態を説明するブロック図である。

【0303】ホストコンピュータから S/P サーバー装置 SP1 へのデータの流は、図 32 に示すように、まず、ホストコンピュータで DTP のアプリケーションプログラム 56 が実行されていてドキュメントを作成する。このドキュメントのデータ形式は、ホストコンピュータの機種、アプリケーションプログラムに依存したものである。そこで S/P サーバー装置 SP1 が解釈できる形式のデータに変換する必要がある。その変換プログラムが変換プログラム 54 である。DTP のアプリケーションプログラム 56 からは DTP アプリに依存したデータが変換プログラム 54 に送られる。変換プログラム 54 は、受けとったファイルを CaPSL コードに変換する。さらに、CaPSL コードに変換したファイルは通信プログラムである lpd505 または通信プログラム 53 に送られる。

【0304】通信プログラムはホストコンピュータと S/P サーバー装置 SP1 の間でネットワークを介して接続、通信を行なうためのものであり、ここでは 2 つのプログラムを用意する。

【0305】先ず始めに、通信プログラム 53 を用いた

時を説明する。

【0306】通信プログラム 53 に対応するプログラムとして、S/P サーバー装置 SP1 側で通信を司るプログラムは通信プログラム 83 であり、このプログラム間では TCP/IP プログラムを使用する。

【0307】プリント時の通信プログラム 53 の役割は主に 2 つあり、一つは下位レイヤ (TCP/IP) を介して指定した S/P サーバー装置 SP1 とリンクを張り、通信できる状態にすることである。もう一つは、変換プログラム 54 が生成した CaPSL データファイルを通信プログラム 83 に送信することである。

【0308】また、通信プログラム 83 の役割はリンクを張りプリント処理に必要な情報をやりとりして最適にプリントできるようにすること。通信プログラム 53 の送ってくるデータを受けとり、システム全体制御プログラムにデータの到着を知らせる。この時、送られてきた CaPSL データ、およびそれに付随する情報は受信バッファに一時的に貯められている。

【0309】また、S/P サーバー装置 SP1 側で何らかのエラーが起こった時に、その情報をホストコンピュータ側の通信プログラム 53 に送信するというような役割もある。ホストコンピュータ側の通信プログラム 53 からプリントアウトする際には、DTP アプリケーションプログラム 56 で作成したドキュメントデータは指定された S/P サーバー装置 SP1 に送られたことになる。次に通信プログラム 83 はシステム全体制御プログラム 93 にジョブ (例えば BJC カラー複写装置からプリントアウト) の到着のイベントを送る。システム全体制御プログラム 93 はイベント駆動型のプログラムで通信プログラム 83、デバイスドライバ 86、PDL インタープリタープログラム 84 から送られてくるイベントを常に待っている。イベントが入ってくるとイベントの発信元と内容を調べてそれに対応した処理を行なう。

【0310】今、システム全体制御プログラム 93 には通信プログラム 83 から「ジョブが到着した」というイベントが入力されている。この時システム全体制御プログラム 93 はイベントを解析してハードディスク 519 を持っている時には受信バッファのドキュメントデータを一旦スプールする。そして、ジョブが幾つかたまっている場合には、ジョブの内容、プリンタ、スキャナの状態、ジョブの優先順位などを考慮して最適にジョブを起動する。

【0311】ハードディスク 519 がいない場合には、データを貯めておくことが出来ないで、データの格納されているアドレスとサイズを PDL インタープリタープログラム 84 に知らせてプリントアウト処理を直ちに行なうように要求を出す。ここでは、ハードディスクがあるものとして説明を続ける。システム全体制御プログラム 93 はジョブの内容を判断して PDL インタープリタープログラム 84 にスプールファイル名を渡し、起動の

要求をする。PDLインタープリタープログラム84はスプールファイルからCaPSLデータを読み込み、解釈して、図形、文字、イメージをバンドメモリ518に描画する。CaPSLデータの中に圧縮符合化されたイメージデータがあった場合にはPDLインタープリタープログラム84は標準圧縮伸長部606でイメージデータを伸長してバンドメモリに描画する。1バンド分の描画を終了すると、PDLインタープリタープログラム84はシステム全体制御プログラム93に対して「1バンド描画終了」のイベントを発行する。イベントを受けとったシステム全体制御プログラム93はデバイスドライバ86に「1バンド排紙」の要求イベントを発行する。

【0312】一方、バンドメモリ518からプリントアウトする場合には、「1バンド排紙」の要求イベントを受けたデバイスドライバ86はインタフェースボード610Bを制御して指定したバブルジェットカラープリンタ（BJプリンタ）656にてプリントを行う。メインCPUボード610のバンドメモリ605に展開されている1バンドの画像データをバンドメモリ653に転送する。バンドメモリ653のデータはBJのヘッドに合った走査形式で読み出されバッファ651を通して画像処理部652に入る。ここでは予め設定したパラメータに従って処理がなされる。通常はバンドメモリにあるNTSC-RGBをBJプリンタ656内部のRGBに変換する処理を行う。そしてインタフェース618を通してBJプリンタ656のプリンタエンジン部へ送られる。BJプリンタ656への制御はCPU601からのコマンドを解釈して、CPU611が行う。なお、650はデュアルポートRAMである。

【0313】最終バンドまでこれらの処理を繰り返したら、PDLインタープリタープログラム84は「ドキュメント終了」イベントをシステム全体制御プログラム93に発行しプリントを終える。プリント終了の際、システム全体制御プログラム93は必要ならば通信プログラム83に「プリント終了」イベントを発行する。通信プログラム83はホストコンピュータ側の通信プログラム

$$\begin{aligned} [Ra] & [a11 \quad a12 \quad a13] [y] \\ [Ga] & = [a21 \quad a22 \quad a23] [cr] \\ [Ba] & [a31 \quad a32 \quad a33] [cb] \dots (1) \end{aligned}$$

ここで、Ra, Ga, Baは色空間Aの任意の1点の座標（Ra, Ga, Ba）であり、対応するYCrCb色空間上の1点を（y, cr, cb）とする。これらのすべての色空間上で近似できるように、例えば最小2乗法により、a11～a33までの係数を求めて利用する。従って、上記第（1）によりデバイスの入力色空間と通信上の色空間に変換を行う。

【0321】一方、通信路上の標準色空間から記録側の色空間に変換する場合、いくつかの方法があるが、本実施例では以下のように処理する。

【0322】まず、通信路上での標準色空間をYCrCb

53に対しプリント終了を知らせる。

【0314】また、エラー発生時には、例えばプリンタ656が、紙詰まりを起こしたり、紙無しの状態になった時には、デバイスドライバ86からシステム全体制御プログラム93に対して「エラー発生」のイベントが送られる。システム全体制御プログラム93はその旨をPDLインタープリタープログラム84と通信プログラム83に伝える。PDLインタープリタープログラム84はプログラムの状態の退避などのエラー時の処理を行ない、通信プリンタ83はエラーの発生や、その内容をホストコンピュータ側の通信プログラム53に伝える。なお、エラーが回復したかどうかを調べる方法は2つ考えられる。一つは、ある一定期間毎にシステム全体制御プログラム93がデバイスドライバ86に対して問い合わせ、デバイスドライバ86が答えるという方法。

【0315】もう一つは、システム全体制御プログラム93がエラー回復を監視していて回復時にシステム全体制御プログラム93に対して「エラー回復」のイベントを発行するというものである。

【0316】以下、異機種間でカラー画像の通信を行う場合、単純にこれらの入出力機器を接続したのではお互いの特性が異なり、最適な色再現が難しくなってしまう。このため、現在では、各デバイス内では、固有の色空間で、通信路上では標準の色空間で通信し合う方向で検討が進んでいる。そこで、以下、このような要請の下での色空間の変換処理方法について説明する。

【0317】なお、説明上、送信側の入力デバイスの色空間をA、通信路上の色空間をB、受信プリンタの色空間をCとする。

【0318】また、通信路上の色空間は現在では比較的知られている色空間、例えばカラー画像符号化で良く用いられるYCrCb色空間の場合について説明する。

【0319】一方、送信側の色空間Aは、YCrCb色空間と異なり色域が異なるのが一般的で、通常は両色空間間を第（1）式のような形式で結合する。

$$\begin{aligned} [0320] \quad [Ra] & [a11 \quad a12 \quad a13] [y] \\ [Ga] & = [a21 \quad a22 \quad a23] [cr] \\ [Ba] & [a31 \quad a32 \quad a33] [cb] \dots (1) \end{aligned}$$

bとした場合、YCrCbからRGBに変換される。YCrCb空間は、NTSCと線形変換できるので、以下、NTSC色空間として説明する。

【0323】また、NTSC標準色空間は加法混色を基本としているが、印刷は減法混色系が用いられる。従って、加法混色と減法混色の変換が必要となる。この変換は、構成が複雑で純粋に論理的に解決するのは非常に難しい。そこで、本実施例では記録側の色空間に近い加法混色系色空間を記録側の内部的標準色空間とする。ここでは、NTSC色空間より、狭いHDTV（High Definition TV）色空間を記録側標準色空間D

とする。

【0324】さらに、記録側内部標準色空間Dとデバイ

$$\begin{aligned} [Y] & [A11 \ A12 \ A13] [Rh] \\ [M] &= [A21 \ A22 \ A23] [Gh] \\ [C] & [A31 \ A32 \ A33] [Bh] \\ [K] & [A41 \ A42 \ A43] \dots\dots (2) \end{aligned}$$

ここで、Y、M、C、Kは印刷のための原色で、イエロー、マゼンタ、シアン、ブラックの各成分である。Rh、Gh、Bhは記録側標準色空間成分である。また、A11～A43までは、印刷と記録内部標準色空間を複数の点で関係付け、最小2乗法で計算される係数である。

【0326】また、通信路上の標準色空間と記録内部の標準色空間は下記のように変換する。

【0327】通信回線上のNTSC標準色空間は基礎刺激からNTSCが表現できる色間の3次元的最外郭面代表情報を求める。同様に、記録側内部標準色空間HDTVの最外郭面の代表位置情報も得られる。

【0328】今、NTSC色空間上の1点(Rn, Gn, Bn)からHDTV色空間の対応点(Rh, Gh, Bh)を求める場合は、Rn, Gn, BnからCIEL*a*b*変換し、Ln, an, bnとする。同様に、Rh, Gh, BhからCIEL*a*b*変換し、Lh, ah, bhとする。Ln一定で $\theta = \arctan(an/bn)$ に近いHDTV、NTSC各色空間上の最外郭近似位置を前述テーブルから求める。そのNTSC最外郭面位置を(Lon, aon, bon)、HDTV最外郭面位置を(Loh, aoh, boh)とすると、ah, bhは下記第(3)式により決定される。

【0329】

$$\begin{aligned} ah &= (aoh/aon) * an \\ bh &= (boh/bon) * bn \dots\dots (3) \end{aligned}$$

これらのah, bhが色空間圧縮後のHDTV色空間上での対応位置である。

【0330】従って、NTSC色空間上の任意の1点(Rn, Gn, Bn)は、記録側標準色空間上では、(Rh, Gh, Bn)に変換される。従って、上述(2)式により印刷すべきY、M、C、Kの各成分量が決定され、印刷可能となる。

【0331】なお、本実施例では記録側内部に標準色空間を設けているが、これは通信回線上での標準色空間は1つに絞られていないためである。従って、複数の標準色空間が使用される可能性がある。その場合でも、本方式を用いていれば通信上の色空間がNTSC色空間でなくともそれ自身の色空間を規定できるものであれば、上記方法により標準色空間変換が可能となる。

【0332】以下、図35を参照しながら周辺機器の異常処理について説明する。

【0333】図35は本発明に係るスキナプリンタサーバーシステムとのネットワークシステムの構成を説明

ス色空間Cとの関係は、下記第(2)で決定される。

【0325】

するブロック図である。

【0334】ジョブ実行中に周辺機器に異常が発生した場合、S/Pサーバー装置SP1664は異常が発生した周辺機器が異常から回復するまで待つのではなく、その状態で実行可能なジョブを優先的に実行していく(実行可能なジョブに関しては、後述する)。また、異常が発生したときに異常が発生した周辺機器に対して実行していたジョブ(以下エラージョブともいう)を、ジョブ再開のために必要な情報を保存した後に、実行時にエラーが発生した処理待ちのジョブとしてS/Pサーバー装置SP1664内に登録することにより、異常が発生した周辺機器が異常から回復した後にジョブを再開することが可能である。エラージョブを再開可能なように登録するか登録しないかは、発生した異常の程度や、SPサーバーの設定による。

【0335】例としてサーバー装置に3台のコンピュータ661～663および2台のカラースキナプリンタ665、666および1台のスキナ667が接続されている場合で説明する。

【0336】カラースキナプリンタ665、666は、各々プリントとスキンのジョブを処理することが出来る。以下、図36に示すフローチャートを参照しながら異常発生ジョブ処理動作について説明する。S/Pサーバー装置664

図36は本発明に係るスキナプリンタサーバーシステムにおける異常発生ジョブ処理手順の一例を示すフローチャートである。なお、(1)～(10)は各ステップを示す。

【0337】例えばホストコンピュータ661がカラースキナプリンタ665に対するプリントジョブをS/Pサーバー装置664に依頼して実行しており、ホストコンピュータ662がシステム全体制御プログラム93にプリントジョブを依頼して処理待ちになっており、ホストコンピュータ663がシステム全体制御プログラム93にスキンジョブを依頼して処理待ちになっていたとする。このプリントジョブ実行中にカラースキナプリンタ665のプリンタ部分に異常が発生したとすると(1)、この時、まずS/Pサーバー装置664はカラースキナプリンタ665に発生した異常が回復困難な異常かどうかを調べる(2)。カラースキナプリンタ665に発生した異常が回復困難な異常であるとS/Pサーバー装置SP1が判断した場合、S/Pサーバー装置SP1は直ちにエラージョブを放棄し(10)、次の処理可能なジョブを選択し実行する(9)。

【0338】なお、本実施例において、処理可能なジョブとは、現在異常が発生しているカラースキャナプリンタ665に対するプリントジョブ以外のジョブ、すなわち、

- ・カラースキャナプリンタ665に対するスキャンジョブ
 - ・カラースキャナプリンタ666に対するプリントジョブ
 - ・カラースキャナプリンタ666に対するスキャンジョブ
 - ・カラースキャナプリンタ667に対するスキャンジョブ
- である。

【0339】処理可能なジョブが複数ある場合は、後述する方式に従ってジョブを実行する。

【0340】カラースキャナプリンタ665に発生した異常が回復困難な異常であるとS/Pサーバー装置664が判断しなかった場合、S/Pサーバー装置664はカラースキャナプリンタ665に対してリトライ処理（図37参照）を行なう。

【0341】リトライは、事前に定めておいたN回（例えば5回）まで繰り返して行なわれる(3)。N回のリトライを行なうまでにカラースキャナプリンタ665が異常状態から回復した（リトライに成功した）場合、ジョブを再開する(6)。N回のリトライを行なってもカラースキャナプリンタ665が異常状態から回復しなかった（リトライに失敗した）場合、S/Pサーバー装置664は他の処理可能なジョブがあるかどうかを調べる(4)。処理可能なジョブがなかった場合、S/Pサーバー装置664はリトライ回数とは別に定めておいたM回（例えば1回）のリトライを行ない(5)、リトライに成功したならば、エラージョブを再開する。リトライに失敗した場合は、再び処理可能なジョブがあるかどうかを調べる(4)。処理可能なジョブがあった場合、S/Pサーバー装置664はエラージョブのジョブ退避を行ない(7)、エラージョブを実行途中でエラーが発生したジョブとして登録し(8)、次の処理可能なジョブを実行する(9)。

【0342】なお、本実施例において、ジョブ退避とは、エラージョブ実行時のS/Pサーバー装置664の状態情報などのジョブ再開時に必要な情報を、例えば記憶手段（例えばハードディスク）や、例えばS/Pサーバー装置内のメモリや、例えばS/Pサーバー装置内のプログラムなどに保存する処理である。

【0343】図37は本発明に係るスキャナプリンタサーバーシステムにおけるリトライ処理手順の一例を示すフローチャートである。なお、(1)～(5)は各ステップを示す。

【0344】リトライ処理とは、事前に定めておいた時間（例えば30秒）待機した後に(1)、異常が発生した周辺機器が異常状態から回復したかどうかを調べ(2)、

ある一定回数X（例えば5回）繰り返す(3)。一定回数繰り返すまでに異常が発生した周辺機器が異常状態から回復していればリトライは成功であり(5)、回復しなければリトライは失敗となる(4)。

【0345】以下、本発明に係るスキャナプリンタサーバーシステムによるマルチプリント制御動作について詳述する。

【0346】スキャナ/プリンタネットワークサーバー（SPネットワークサーバー）には、スキャナとプリンタとから構成されるカラーレーザビーム複写装置（CLC）、スキャナとプリンタとから構成されるバブルジェットカラー複写装置（BJC）、標準インターフェースで接続されているビットマッププリンタ、PDLプリンタ（CaPSL、PS）、市販されるスキャナ等多くのデバイスが接続可能になっている。また、ネットワーク上のホストマシンからは、これらのデバイスを自由に使用させるため、以下のように各ジョブを制御している。

【0347】例えばホストコンピュータAからカラーレーザビーム複写装置に10頁のドキュメントを出力する要求がSPネットワークサーバーに送られ、さらにホストコンピュータBからバブルジェットカラー複写装置（BJC）のスキャナからカラー原稿を入力する要求が発生し、さらにホストコンピュータCからPDLプリンタに出力するという要求が発生するといった事態も想定される。このように、ネットワーク上のホストコンピュータからは、様々な要求がSPネットワークサーバーに送られ、SPネットワークサーバーでは、それらの要求（ジョブ）に対処するべく下記のようなジョブ制御を実行する。なお、本実施例ではプリント要求、スキャン要求をジョブと呼ぶ。例えばカラーレーザビーム複写装置（CLC）のプリンタにCaPSLコードで記述された3頁のドキュメントを出力する要求を1つのジョブと考える。SPネットワークサーバーのジョブ制御では、同時に2つまでのジョブしか走らないようにするが、3つ以上のジョブが走る場合も制御を拡張することによりほぼ同様に制御できる。

【0348】以下、図38を参照しながら本発明に係るスキャナプリンタサーバーシステムにおけるジョブ制御動作について説明する。

【0349】図38は本発明に係るスキャナプリンタサーバーシステムにおけるジョブ制御状態を説明するタイミングチャートである。なお、ジョブの処理は、SPネットワークサーバーがスプール用のハードディスクを備えているかどうかにより、図38の(a)～(e)の何れかのジョブ制御1～5に大別される。また、説明上ジョブ1、2は3頁のプリント要求または3頁のスキャン要求とする。

【ジョブ制御1】ジョブ制御1は、同図(a)に示すように、先に要求のあったジョブを実行して、次のジョブ2は、ハードディスクにスプールして、ジョブが終了し

た時点でジョブ2を起動する。

〔ジョブ制御2〕ジョブ制御2は、同図(b)に示すように、特殊な場合であるが、1つのジョブで2つ以上のデバイスを動かす場合の制御に対応する。例えばメモリに展開した画像をカラーレーザビーム複写装置(CLC)、バブルジェットカラー複写装置(BJC)にも出力する場合である。

〔ジョブ制御3〕ジョブ制御3は、同図(c)に示すように、ジョブの中にページという概念をいれる。この場合、プリントの場合にはドキュメントのページ単位、スキャナの場合には1ページの原稿のスキャンの単位とする。例えばジョブ1を実行中にジョブ2の要求がきた場合には、ジョブ1をページの切れ目で中断して、ジョブ2を行う。この場合、ジョブ1とジョブ2とは同時に動いていることはない。

〔ジョブ制御4〕ジョブ制御4は、同図(d)に示すように、ジョブ1を実行中にジョブ2の要求がきた場合は、ジョブ1を実行したまま、直ちにジョブ2を実行する。

〔ジョブ制御5〕ジョブ制御5は、同図(e)に示すように、ジョブ1を実行中にジョブ2の要求がきた場合は、ジョブ2をリジェクトする。

【0350】なお、図1に示したサーバー装置の構成の場合には、ジョブの種類が下記の(1)～(7)に分類される。

【0351】(1) カラーレーザビーム複写装置(CLC)のプリンタにドキュメントを出力する。

【0352】(2) カラーレーザビーム複写装置(CLC)のスキャナから原稿を読み取る。

【0353】(3) バブルジェットカラー複写装置(BJC)のプリンタにドキュメントを出力する。

【0354】(4) バブルジェットカラー複写装置(BJC)のスキャナから原稿を読み取る。

【0355】(5) 市販のページ記述言語のインタプリタを内蔵するプリンタに出力する。

【0356】(6) 市販のビットマッププリンタにドキュメントを出力する。

【0357】(7) 市販のスキャナから原稿を読み取る。

【0358】以下、各ジョブの優先順位処理について説明する。

【0359】例えばスキャンのジョブを至急行いたい時、プリントジョブが実行中ならば、プリントジョブを中断して、スキャナジョブを行うことができるように、本実施例では優先順位0～2の3段階の指定可能であり、優先順位0が指定なし(ファーストインファーストアウト)の場合、優先順位1がジョブ単位で優先割込み指定ありの場合、優先順位2がページ単位で優先割込み指定する場合である。

【0360】以下、図39を参照しながら本発明に係る

スキャナプリンタサーバーシステムにおけるSPマネージャー制御動作について説明する。

【0361】図39は本発明に係るスキャナプリンタサーバーシステムにおけるシステム全体制御プログラム93の制御処理状態を示す図である。本実施例において、システム全体制御プログラム93は、主としてホストコンピュータから送出されてくるジョブの優先順位、ジョブの種類を参照してプリンタのジョブを最適に制御する。以下、図面に基づいて、SPネットワークサーバに対して、ネットワーク上のホストコンピュータから様々な要求が送出される場合のジョブ管理について説明する。

【0362】まず、各プログラムの概念的な構成を説明すると、図39に示すように、通信に関する通信プログラム83、SPネットワークサーバーの全体を制御するシステム全体制御プログラム93、スキャナに関するプログラムのスキャナ制御プログラム85、プリント処理(CaPSLのインタプリタ)に関するプログラムのPDLインタープリタープログラム84、入出力機器を実際に制御するデバイスドライバ86の5つのプログラムに大きく分けられる。さらに、実際にデータが流れるインタフェースとして、バンドメモリ(BMEM)70-1、セントロニクスインタフェース(CENTR)70-2、RS232Cインタフェース(RS232)70-3があり、これらをシステム全体制御プログラム93が管理する構成となっている。

【0363】システム全体制御プログラム93は基本的に他の4つのプログラムからのイベントにより動く、イベント駆動型の処理を図40に示すフローチャートに従って実行する。

【0364】図40は、図39に示したシステム全体制御プログラム93のイベント処理手順の一例を示すフローチャートである。なお、(1)～(9)は各ステップを示す。

【0365】まず、SPネットワークサーバーの電源投入時にシステム全体制御プログラム93は起動し、すぐにイベント待ちの無限ループになる。プログラムが走り出すと、イベント待ちの状態になる(1)。イベントが入力されるとループから抜け出し、もし、そのイベントが通信プログラム83からのものかどうかを判定し(2)、YESならばイベントに応じた処理を行なう(3)。そして、またステップ(1)に戻りイベント待ちのループになる。

【0366】一方、ステップ(2)の判定でNOの場合は、そのイベントの発行先がPDLインタープリタープログラム84のプリントジョブかどうかを判定し(4)、YESならばイベントに応じた処理を行なう(5)。

【0367】一方、ステップ(4)の判定でNOの場合は、そのイベントの発行先がスキャナ制御プログラム85のスキャナジョブかどうかを判定し(6)、YESなら

イベントに応じた処理を行なう(7)。

【0368】一方、ステップ(6)の判定でNOの場合は、そのイベントの発行先がデバイスドライバ86からかどうかを判定し(8)、NOならばステップ(1)に戻り、YESならばイベントに応じた処理を行い(9)、ステップ(1)に戻る。

【0369】一方、通信プログラム83からは、カラーレーザ複写装置(CLC)のプリントに関するジョブや、バブルジェットカラー複写装置のスキナに関するジョブ等、いろいろなジョブの要求が定期的にシステム全体制御プログラム93に送られる。それら複数のジョブを資源の許す限り最適に振り分けなければならない。以下そのアルゴリズムについて図41を参照しながら説明する。

【0370】図41は本発明に係るスキナプリンタサーバーシステムにおけるジョブの管理状態を示す図である。

【0371】この図に示されるように、ジョブの管理にはジョブテーブルを使用する。ジョブテーブルには、ジョブを識別するためのID、ジョブの状態を示すステータス、ジョブの実行の優先順位、ジョブの種類、終了ページ、そして割り込みジョブIDがある。

【0372】IDはシリアル番号で、ジョブの要求の受け付け順と考えて良い。ステータスには、実行中を示すRUN、処理を待っているWAIT、割り込みが入って中断していることを示すSTOP、エラーが発生して回復待ちを示すESTOPがある。優先順位はLEVEL0から2までであり0は「指定なし」で通常はこのレベルを指定する。1は「ジョブ単位で優先割り込み」で幾つかのジョブが待ち状態の時、それらのジョブより優先して処理するというものである。また2は「ページ単位で割り込み」を表し、現在ジョブを処理中でも、ページの切れ目であればそのジョブを中断してLEVEL2のジョブを処理する。

【0373】インタフェースは、図39に示すようにそのジョブが使用するハードウェアを指し、ジョブ同士が排他的にハードウェアを利用可能にするために設けたものである。また、ジョブはカラーレーザ複写装置(CLC)のプリンタからドキュメントを出力するCLCP、カラーレーザ複写装置(CLC)のスキナから原稿を読みとるCLCS、バブルジェットカラー複写装置のプリンタにドキュメントを出力するBJP、バブルジェットカラー複写装置のスキナから原稿を読みとるBJS、市販のページ記述言語のインタブリタを内蔵するプリンタに出力するPDLP、市販のビットマッププリンタにドキュメントを出力するBITP、市販のスキナから原稿を読みとる等のジョブがある。なお、終了ページにはジョブが処理し終ったページ数を記録しておく。これは、もしエラーが起こった時にエラー回復後、どのページから処理を再開すれば良いかの判断に使われる。

また最後の割り込みジョブIDは、割り込んだジョブが終了しているかの判断に使われ、終了していれば、STOP中のジョブを再開する。

【0374】システム全体制御プログラム93は以上のような情報の格納されたジョブテーブルを参照しながら、次にどのジョブを起動するかを最適に判断する。

【0375】以下、図42に示すフローチャートを参照しながら本発明に係るスキナプリンタサーバーシステムにおけるジョブ実行処理動作について説明する。

【0376】図42は本発明に係るスキナプリンタサーバーシステムにおけるジョブ実行処理手順の一例を示すフローチャートである。なお、(1)～(6)は各ステップを示す。

【0377】なお、このフローに制御が移るタイミングは、次のようになる。1. システム全体制御プログラム93にイベントが入らずアイドル状態の時にある一定の間隔をおいてこのフローに移る。そして、ジョブテーブルにジョブがなければまたアイドル状態になる。2. 通信プログラム83からのイベントがあった時。3. PDLインタープリタプログラム84、スキナ制御プログラム85から、ページ終了のイベントまたはドキュメント終了のイベントがあった時。4. デバイスドライバ86からエラー等のイベントがあった時。

【0378】先ず、ステップ(1)において、図41に示すジョブテーブルを参照する。次いで、ステップ(2)においてジョブがあるかどうかの判断をする。実行すべきジョブがない時にはステップ(7)に移り、システム全体制御プログラム93はアイドル状態になる。実行すべきジョブがある場合にはステップ(3)で実行可能なジョブの候補を幾つか選び、ステップ(4)で候補に上がったジョブの優先順位を見て一つのジョブに絞る。さらに、ステップ(5)でその絞り込まれたジョブが使用するインタフェースが空いているかを判断する。ステップ(6)で実際にジョブを実行し、処理を終了する。

【0379】以下、図43に示すフローチャートを参照しながら本発明に係るスキナプリンタサーバーシステムにおけるステータスチェック処理動作について説明する。

【0380】図43は本発明に係るスキナプリンタサーバーシステムにおけるステータスチェック処理手順の一例を示すフローチャートである。なお、(1)～(14)は各ステップを示す。

【0381】先ず、ステップ(1)でジョブテーブルから一つのジョブのステータスを読みとる。ステップ(2)でジョブのステータスがRUNかどうかを判断し、RUNならばステップ(14)に移りジョブテーブルのジョブをすべて読みとったかを判断する。もし、まだジョブが残っていればステップ(1)で次のジョブのステータスを読みに行く。RUNでなければステップ(3)に移る。ステップ(3)ではジョブのステータスがWAITかどうかの判

断をする。WAITならばステップ(4)で実行可能なジョブの候補としてピックアップする。そしてステップ(14)に移る。WAITでない場合には、ステップ(5)に移り今度はステータスがESTOPかどうかの判断をする。もし、このジョブのステータスがESTOPならば、エラーを起こして中断しているため、ステップ(6)でこのジョブを中断した原因のエラーが回復しているかどうかのチェックを行なう。ステップ(7)はエラー中か、そうでないかの判断の分岐になる。まだエラーの場合には実行可能なジョブの候補にはピックアップせずにステップ(14)に移る。エラーが回復している場合には、そのジョブのステータスをESTOPからWAITにジョブテーブルを書き換える。そしてステップ(9)で実行可能なジョブの候補としてピックアップする。またステップ(5)でステータスがESTOPではないと判断された場合には、他のジョブに割り込まれて中断しているSTOPの状態なので、ステップ(10)で割り込んだジョブのステータスをチェックする。ステップ(11)では割り込んだジョブが実行中かどうかの判断をする。実行中ならば実行可能なジョブの候補にはピックアップせずにステップ(14)に移る。もし、実行中でなければステップ(12)に移り、そのジョブのステータスをSTOPからWAITにジョブテーブルを書き換える。そして、ステップ(13)で実行可能なジョブとしてピックアップする。最後に、すべてのジョブをジョブテーブルから読み終わったら(14)、ステータスのチェックは終了する。このようにして実行可能なジョブの候補が複数選ばれる。

【0382】以下、図44に示すフローチャートを参照しながら本発明に係るスキャナプリンタサーバーシステムにおける優先順位チェック処理動作について説明する。

【0383】図44は本発明に係るスキャナプリンタサーバーシステムにおける優先順位チェック処理手順の一例を示すフローチャートである。なお、(1)～(11)は各ステップを示す。また、本実施例では候補に上がったジョブの優先順位を見て実行するジョブを一つに絞るということをする。

【0384】ステップ(1)で実行可能なジョブの候補としてあげられた複数のジョブの内一つに対してその優先順位をリードする。ステップ(2)でLEVEL2かどうかの判断をする。LEVEL2ならばステップ(3)に移りLEVEL2のジョブが存在するかどうかを表すフラグFLAG2を立てる。LEVEL2でなければステップ(4)に移り今度はLEVEL1であるかの判断をする。LEVEL1ならばステップ(5)に移りLEVEL1のジョブが存在するかどうかを表すフラグFLAG1を立てる。もしLEVEL1でもないかと判断された場合には、LEVEL0になる。すなわちFLAG2もFLAG1も立っていない場合にはLEVEL0になる。

【0385】一つのジョブの優先順位の判断が終るとス

テップ(6)に移り候補に上がったジョブをすべて調べたかどうかを判断する。まだすべて調べ終わっていない時にはステップ(1)に戻り、次のジョブの判断に移り、調べ終わった場合にはステップ(7)に移る。フラグFLAG2が立っている時にはステップ(8)に移り、優先順位がLEVEL2でジョブのIDが小さい(早くジョブのリクエストを受け付けたもの)ジョブを選択する。

【0386】同様に、ステップ(9)、(10)では優先順位がLEVEL1でジョブのIDが小さいジョブを選択する。ステップ(11)では優先順位がLEVEL0でIDの小さいジョブが選択され、優先順位のチェックが終了する。

【0387】このようにして次に起動可能なジョブを一つに絞る。起動可能なジョブが決定しても、インタフェースの状態により実際に起動できるかどうかが決まってくるのでジョブとインタフェースの関係を調べなくてはならない。

【0388】以下、図45に示すフローチャートを参照しながら本発明に係るスキャナプリンタサーバーシステムにおけるインタフェースチェック処理動作について説明する。

【0389】図45は本発明に係るスキャナプリンタサーバーシステムにおけるインタフェースチェック処理手順の一例を示すフローチャートである。なお、(1)～(9)は各ステップを示す。

【0390】まず、ステップ(1)において、一つに絞られたジョブのインタフェースをジョブテーブルからリードする。ステップ(2)でそのインタフェースの状態をチェックする。使用中でなければステップ(5)に移りそのジョブを実行ジョブテーブルに登録する。実行ジョブテーブルは、実際に起動するジョブを表すテーブルでマルチでジョブを走らせることが出来るので複数のジョブが登録可能である。

【0391】ステップ(6)のジョブ処理ではこの実行ジョブテーブルを参照してジョブを起動する。

【0392】インタフェースが使用中の場合にはステップ(3)に移りジョブの優先順位をチェックする。LEVEL2でなければ、現在のジョブを中断させることは出来ないため実行ジョブテーブルに登録することはしないで、処理を終了する。また、ステップ(3)でLEVEL2の時には、ステップ(4)で現在ステータスがRUNのジョブがLEVEL2であるかの判断をする。もしRUNのジョブがLEVEL2であるならば中断させることは出来ないため、選択されたジョブは実行ジョブテーブルに登録せずに、処理を終了する。

【0393】一方、現在実行中のジョブ(ステータスがRUN)の優先順位がLEVEL1またはLEVEL0の場合には現在実行中のジョブを中断させて新たに選択されたジョブを起動させる。その場合、ステップ(6)に移り、ステップ(4)で選択されたジョブを実行ジョブテ

ーブルに登録する。ステップ(7)で現在実行中のジョブのステータスをSTOPにして、さらに優先順位をLEVEL2にする。これは、ジョブが復帰した時に優先して実行されるようにするためである。

【0394】ステップ(8)で現在実行中のジョブが何ページまで処理したかを記録しておくために終了ページ数をジョブテーブルに書き込む。さらにステップ(9)で現在実行中のジョブを実行ジョブテーブルから削除する。これで、現在実行中のジョブが起動されることはなく、ステータスもSTOPになり再度ジョブが起動されるのを待機することになる。このようにしてインタフェースチェック処理を終了する。

【0395】以下、図46に示すフローチャートを参照しながら本発明に係るスキナプリンタサーバシステムにおけるジョブ起動処理動作について説明する。

【0396】図46は本発明に係るスキナプリンタサーバシステムにおけるジョブ起動処理手順の一例を示すフローチャートである。なお、(1)～(4)は各ステップを示す。

【0397】まず、ステップ(1)において、実行ジョブテーブルを参照して起動すべきジョブをリードする。ステップ(2)でこれから起動するジョブがステータスESTOPでエラーストップしているジョブの種類とどうかをチェックする。もし同じであれば、起動することは出来ないでステップ(4)に移り、実行ジョブテーブルをすべて読んだかを判断する。もしまだ残っていればステップ(1)に移り次のジョブの起動処理をする。もしエラーストップしているジョブの種類と違っていれば実行可能なので、ステップ(3)に移りジョブの起動に移る。実際のジョブを起動する時は、ステータスをWAITからRUNに替え、インタフェースを確保する。逆にエラーストップ、または割り込みで中断する時には、インタフェースを解放する。

【0398】ステップ(4)で実行ジョブテーブルにジョブが残っているかを判断して、もう起動すべきジョブが残っていなければ、ジョブ起動処理を終了する。

【0399】以下、図47～図57を参照しながら本発明に係るスキナプリンタサーバシステムにおける具体的ジョブ処理動作について説明する。

【0400】図47は本発明に係るスキナプリンタサーバシステムにおけるジョブ処理状態推移を示すタイミングチャートである。なお、図中の横軸右方向に時間が流れていくものとする。図において、801～813はジョブテーブルをチェックするタイミングを示し、タイミング800～803は通信プログラム83からジョブのイベントがシステム全体制御プログラム93に入るタイミングに対応し、タイミング804～813はジョブテーブルをチェックするタイミングに対応する。

【0401】図48～図57は本発明に係るスキナプリンタサーバシステムにおけるジョブ処理に伴うジョ

ブテーブルの内容を示す図であり、図48～図57はタイミング804～タイミング813におけるジョブテーブルに対応する。なお、ジョブテーブルはチェック直前の内容を示している。また、ジョブはカラーレーザ複写装置(CLC)のプリンタからドキュメントを出力するCLCP、カラーレーザ複写装置(CLC)のスキナから原稿を読みとるCLCS、バブルジェットカラー複写装置のプリンタにドキュメントを出力するBJP、バブルジェットカラー複写装置のスキナから原稿を読みとるBJS、市販のページ記述言語のインタプリタを内蔵するプリンタに出力するPDLF、市販のビットマッププリンタにドキュメントを出力するBITP、市販のスキナから原稿を読みとる等のジョブがある。

【0402】タイミング800で、ジョブのイベントが入ってくる。内容はカラーレーザ複写装置(CLC)のプリンタにドキュメントを出力するジョブ1(内容はCLCP)で、出力ページ数は3ページ、優先順位はLEVEL0、インタフェースにはBMEMを用いるというものである。ジョブテーブルJOBTを見てみると、図48に示すようにIDに「25」が割り当てられ、ステータスはWAIT、終了ページはまだこのジョブは処理されていないので0ページになっている。

【0403】タイミング804のジョブテーブルチェックでは、ID25のジョブが選択され、起動される。この時ID25のジョブのステータスをRUNに変更する。801のタイミングで新たなジョブが入ってくる。ジョブID25のジョブの1ページの処理が終了するとシステム全体制御プログラム93は再度ジョブテーブルを見に行く。その時のジョブテーブルJOBTは図49に示される。ID25のジョブは1ページの出力が終了して、次ページが処理されるのを待っている状態である。終了ページは0ページから1ページに変化している。また、新たに入ったジョブはID26が割り当てられ、ステータスはWAIT、優先順位はLEVEL1、インタフェースはBMEM、ジョブはBJPになっている。システム全体制御プログラム93はこのジョブテーブルJOBTから判断して、ID25のジョブの2ページ目の処理を起動する。

【0404】この時、2ページめの途中でプリンタに紙ジャムが発生してしまいID25のジョブを続けられなくなってしまった。システム全体制御プログラム93はジョブテーブルJOBTを参照して次に起動すべきジョブを探す。その時のジョブテーブルJOBTは図50のようになっている。ID25のステータスはESTOPになり、優先順位は復帰した時に優先的に実行させるためにLEVEL2、終了ページは1ページのままだになっている。そこで、システム全体制御プログラム93はID26のジョブを起動する。ID26のジョブが1ページの処理を終了すると、システム全体制御プログラム93はジョブテーブルJOBTチェック807に移る。こ

の時の状態は図51に示される。ID25のジョブはエラーチェックをしてもエラーが回復していないので起動されることはない。そこで、システム全体制御プログラム93はID26のジョブの2ページ目を処理することをPDLインタープリタープログラム84に対して要求する。

【0405】ID26のジョブが2ページ目を処理している最中に新たなジョブのイベントがシステム全体制御プログラム93に入ってくる。内容は、バブルジェットカラー複写装置のスキナから2ページ分の原稿を読み取るものである。ID26のジョブが2ページ目の処理を終了すると、システム全体制御プログラム93はジョブテーブルJOBTをチェックする。その時のジョブテーブルJOBTが図52である。

【0406】ID25はエラーが回復していないのでそのままの状態である。ID26のジョブは2ページの処理が終了したので、終了ページは2に変化している。さらに新しいジョブがID27に割り当てられている。ID27のジョブは優先順位がLEVEL2であるので、システム全体制御プログラム93は現在処理中のID26のジョブを中断させてID27のジョブを起動する。ID26のジョブはステータスがSTOPに、優先順位がLEVEL2になる。

【0407】スキナ制御プログラム85が1ページの原稿読み込みしている最中に、システム全体制御プログラム93には新たなジョブのイベントがタイミング803で入ってくる。1ページの読み込みが終了すると、システム全体制御プログラム93はタイミング809でジョブテーブルJOBTをチェックする。この時のジョブテーブルJOBTは図53のようになる。ID25のジョブはエラーが回復していないのでESTOPのままである。また、ID26はステータスがSTOPで、割り込みジョブID27であるので、ID27のジョブをチェックするとステータスはRUNのままなので再開することは出来ない。新たに入ってきたジョブはID28が割り当てられる。ID28のジョブはセントロニクスインタフェースを通して市販のプリンタにデータを流すジョブで、現在起動されているジョブと同時に走らせることが出来る。そこで、システム全体制御プログラム93はID28のジョブとID27の2ページ目をスキャンするジョブを同時に起動する。

【0408】次にシステム全体制御プログラム93はジョブテーブルJOBTチェック(タイミング810)に移る。この時のジョブテーブルJOBTは図54に示される。ジョブテーブルJOBTチェック(タイミング810)の前にジョブID25のエラーが解除されているので、エラーチェック後ステータスはWAITに戻る。

【0409】また、ID26のジョブも、割り込んだID27のジョブが終了したのでステータスがWAITに戻る。ID28のジョブはRUNの状態のままである。

システム全体制御プログラム93はインタフェースBMEMが空いているのでID25またはID26のジョブを起動させる。どちらのジョブともステータスはWAITで、優先順位もLEVEL2と状態は同じであるので先に受け付けたID25のジョブを起動する。この時、ID25のジョブは終了ページが1ページであるので2ページ目から処理するようにPDLインタープリタープログラム84に要求を出さなければならない。

【0410】ジョブテーブルJOBTチェック(タイミング811)では図55に従う。ID25とID28のジョブはステータスがRUNでID26はWAITである。ID26のジョブは優先順位がLEVEL2で、現在起動中のジョブに対して割り込むことが出来るがID25もLEVEL2で起動されたので、この場合には割り込むことは出来ない。

【0411】ジョブテーブルJOBTチェック(タイミング812)に移る。ジョブテーブルJOBTは図56に示される。ID25のジョブはすでに終了してジョブテーブルJOBTから削除されている。そこで、システム全体制御プログラム93はID26のジョブを3ページ目から起動する。

【0412】ジョブテーブルJOBTチェック(タイミング813)では、図57に示すようにID28のジョブが走っている状態なので、新たにシステム全体制御プログラム93はジョブを起動することはしない。

【0413】以上のように、システム全体制御プログラム93はジョブテーブルJOBTを参照しながら最適にジョブ管理を行なう。

【0414】なお、上記実施例では文字、図形、イメージがホストコンピュータで編集された後、ネットワークを介して通信されたプリント要求をスキナ/プリンタをドライブするサーバー装置上で処理して、当該プリンタから出力する場合について説明したが、図58に示すように、クライアントホストコンピュータ901、902、スキナプリンタ904をドライブするSPサーバー装置903がネットワーク905を介して接続されるシステムにおいて、クライアントホストコンピュータ901、902から転送された文字、図形の各情報とスキナプリンタ904のスキナから入力されたイメージとをSPサーバー装置903が合成編集することにより、合体出力させるように構成しても良い。

【0415】図58は本発明の第2の実施例を示すサーバー装置の構成を説明するブロック図である。

【0416】図において、906は記憶装置で、SPサーバー装置903内に設けられ、クライアントホストコンピュータ901、902から転送された文字、図形の各情報を蓄える。

【0417】このように構成されたサーバー装置において、本体またはネットワーク上に設けた記憶手段(記憶装置906)に画像処理装置から出力される第1の画像

情報または所定のネットワークに接続される各ホストコンピュータから転送される第2の画像情報を記憶させることにより、各ホストコンピュータからの画像情報出力処理負担を軽減させるとともに、記憶した画像情報を効率よく再利用させる。

【0418】また、記憶手段（記憶装置906）に記憶された第1および第2の画像情報を画像合成手段が合成しながら出力画像を生成して画像処理装置（本実施例ではスキャナプリンタ904）に出力することにより、各ホストコンピュータ上での画像編集負担を軽減させる。

【0419】なお、本実施例においては、クライアントホストコンピュータ901、902上では、例えばワードプロセッサ、グラフィックソフトにより、オペレータが文字情報、図形情報の生成を行う。生成された文字、図形情報等（第1の画像情報）は、ネットワーク905を経由して、SPサーバー装置903に送られる。SPサーバー装置903で受け取った文字、図形情報は、SPサーバー装置903内の記憶装置906に記憶される。

【0420】一方、スキャナプリンタ904でスキャンされた画像情報（第2の画像情報）も、SPサーバー装置903内の記憶装置906に記憶される。SPサーバー装置903上で動作しているページ記述言語により、SPサーバー装置903の記憶装置906内で文字情報、図形情報、画像情報（イメージデータ）が、図59に示すように合成される。

【0421】図59は、図58に示したSPサーバー装置903により合成される印刷レイアウトを示す模式図である。

【0422】図において、911は1ページの領域を示し、この領域911内に、クライアントホストコンピュータ901、902からネットワーク905を介して転送された文字、図形等の転送情報912およびSPサーバー装置903が制御するスキャナプリンタ904でスキャンされた画像情報913がレイアウト情報に従って割付けられる。このように合成された情報がSPサーバー装置903のプリンタから出力させることも可能となる。

【0423】なお、上記SPサーバー装置903に制御されるスキャナプリンタ904は、スキャナとプリンタとが独立した単体構成であっても良い。また、SPサーバー装置903に別の大容量記憶装置、例えば光磁気ディスク装置を接続した場合、スキャナプリンタ904のスキャナで読み取った画像情報をその都度SPサーバー装置903内の記憶装置906に読み込み、クライアントホストコンピュータ901、902から転送された文字、図形情報とを合成して印刷するのではなく、スキャン画像を上記大容量記憶装置内に順次記憶させて行くことにより、画像データベースを構築することで、任意に大容量の記憶装置内から画像を取り出し、その画像を文

字、図形情報と合成しながら印刷させることもできる。さらに、上記大容量の記憶装置をSPサーバー装置903の記憶装置906と別に設ける際、当該大容量の記憶装置とSPサーバー装置903とが直接接続されていなくても良く、例えばネットワーク905に接続されるクライアントホストコンピュータ901、902に配置され、これらのクライアントホストコンピュータ901、902がSPサーバー装置903に転送できる構成であれば良い。

【0424】図60は、図2に示した第1のステーションST1のプログラム構成を説明する図である。

【0425】この図において、DTPアプリケーション1500（図2に示したアプリケーション56に対応する）から印刷しようとする、OSの一部のプリントマネージャー1501（プリントマネージャー）がコールされる。プリントマネージャー1501は、プリンタドライバ1502に対し、描画メモリ領域を確保させそのポインタを提供させる。プリントマネージャー1501は、そのポインタをアプリケーション1500に知らせ、そこにクイックドロー（Quick Draw）フォーマットで描画させる。従って、アプリケーション1500が描画を完了すると、そのデータはすっかりプリンタドライバ1502に渡されている事になる。プリンタドライバ1502は、そのクイックドロー関数を1個1個、順番にCaPSLコードに変換してメモリに蓄え、S/Pクライアント（S/P Client）1503にそれをスキャナ／プリンタサーバー（S/Pサーバー）1602側に伝送する様に依頼をする。S/Pクライアント1503（図2に示した通信プログラム53に対応する）とS/Pサーバー1602（図2に示した通信プログラム83に対応する）との間は、データ転送プロトコルによって結ばれており、プリンタドライバ1502から渡された、変換済みのCaPSLコードをS/Pサーバー1602に転送する。なお、1504は通信プロトコル（図2に示したTCP/IPプログラム52に対応する）、1505はイーサネットボード、1506は圧縮（ADCT）制御関数、1507は圧縮ボードである。また、1600は指定されたプリンタ、1601はCaPSLインタプリタ（図2に示したPDLインタープリタープログラム84に対応する）、1603は通信プロトコル（TCP/IPプログラム82）、1604はイーサネットプロトコルである。

【0426】図61は、図60に示した第1のステーションST1のOSと各ドライバとの関係を示す詳細図であり、図60と同一のものには同じ符号を付してある。

【0427】この図に示すように、ドライバとマッキントッシュで構成されるステーションST1のMac OSの関係を示す。

【0428】図において、1508はQuick Draw／CaPSL変換ルーチン（プログラム54）で、Q

QuickDrawの描画関数をCaPSLコードに変換する。

【0429】デバイスマネージャー、プリントマネージャーをアプリケーションとしてではなく、ドライバとして構成する。インターフェースがMac用のOSを介してつながるので、スペックが標準化しやすい。また、マルチファインダーとは言っても、事実上シングルタスクのMac用のOSの場合、ドライバ化する事によって、その部分をマルチタスク化することができる。UNIXマシンの場合には、文字どおり物理デバイスとのインターフェースのみをドライバ化し、あとはデーモンとして簡単にバックグラウンドで走らせる事ができる。

【0430】また、プリンタドライバ機能は、QuickDraw/CaPSL変換ルーチン1508（図2に示した変換プログラム54に対応する）、プリンタドライバ1502に分かれ、QuickDrawの全ての描画関数をCaPSLコード（の組み合わせ）に置き換える。

【0431】そして、プリンタドライバ1502がLaserWriterがプリントマネージャー1501に対して提供しているのと同じサービスを提供する。

【0432】具体的には、アプリケーション1500が、PrStIDialog()やPrJobDialog()等の関数を発行した時に、プリントマネージャー1501から要求されるパラメーターを返したり、SPClientドライバのコマンドを制御する。

【0433】図62は、図60に示したプリンタマネージャー1501とプリンタドライバ1502との機能処理を説明する図である。

【0434】なお、プリンタドライバ1502は、LaserWriterと同じドラフトモードで動作する。また、プリンタドライバ1502は、プリントマネージャー1501によって起動され、プリントマネージャー1501に対しプリンタドライバインターフェースを提供する。プリンタドライバ1502はプリントマネージャーに対し以下のような関数を提供する。

【0435】Open (SPClientファイルのDRVR 28 リソースをメモリにロックし、OpenConnを起動する)

Close

Read (原稿をスキャンする)

Write (CaPSLコードを印刷する)

Status (SPサーバーの状態をチェックする)

Control (SPサーバーを制御する)

KillIO

また、プリンタドライバ1502の基本はドライバであるが、[Chooser Document]の形で提供し、LaserWriterやImageWriterとともに、Chooser DAから選択可能にする。また、プリンタドライバ1

502は、SPクライアント1503を介してプロトコル(MacTCP)1504の通信機能を利用する。SPクライアント1503は、プリンタドライバ1502に対し標準のドライバ形式のインターフェースを提供する。

【0436】なお、SPクライアント1503のドライバは、初期化処理(INIT-31)のメカニズムによって電源の立ち上げ時に自動的に自分自身をSystemにインストールする。ドライバのメモリー占有サイズが大きくなる時は、大部分のコードはコードリソースの形で持ち、オープンされた時にリソースをSystemヒープにロードする。その場合、Close時にメモリーを解放する。またSPClientは[コントロールパネルドキュメント]のリソースも有し、コントロールパネルから、IPアドレス等の各種パラメーターを設定できる。そのため少なくとも次のcdevリソースを持っている。

【0437】DITL ID = -4064

mach ID = -4064

nrct ID = -4064

ICN# ID = -4064

BNDL ID = -4064

FREF ID = -4064

cdev ID = -4064

また、標準ドライバインターフェースの無いSPクライアント1503のサービスは、Controlルーチンが提供する。Controlコールのパラメーターブロックの持つcsCodeを所定の値に設定する事によって、次の様な各種のコマンドが利用できる。以下、SPクライアント1503とSPサーバー1602とのプロトコルについて説明する。

【0438】SPクライアント1503は、直接的にはTCPプロトコル1504に全てのコマンドとデータを書き込むが、概念上はLAN96を介して接続されたSPサーバー1602のレイヤーと交信している様に振舞う。従って、ここにEnd-to-Endのプロトコルが必要になる。サーバーとクライアントは、互いにパケットをやり取りする事によって、上位レイヤーに通信サービスを提供する。パケットは[サーバー/クライアントパケット]と[ドキュメント制御パケット]に分かれる。これは丁度、OSIのセッションレイヤーとドキュメントレイヤーの関係に依っている。違いは、OSIの様に明確に2つのレイヤーに分けるのではなく、ひとつのレイヤーの中で、パケットの使い方を整理したという点が異なる。

【0439】なお、MacTCPのアプリ側のインターフェースはTCPとUDPが用意されており、プリンタドライバはTCPプロトコルを使う。MacTCPはデバイスドライバである。また、本実施例において、ADCTは、Adaptive Discrete Cosine Transformの略で、JPEGという中間調画像の圧縮技術の核をなすものである。JPEGは、Joint Photographic Expert Groupの略で、CCITTの下部組織(スタンディグループ)である。この組織が発行する中間調画像

の圧縮技術を、JPEG方式と言う。

【0440】以下に、図2に示したステーションST2（IBM社製のパーソナルコンピュータ）をクライアントホストとした場合とSPサーバー1602とのデータ通信処理状態について説明する。

【0441】先ず、Windows環境に於けるネットワークプログラムの特徴について説明する。

【0442】アプリケーションプログラムが印刷しようとする時、従来のOS（MS-DOS）環境ではアプリケーション自身が各社のプリンタに固有の制御コードを出力していた。それが、Windows3.0（商品名）では、アプリケーションとプリンタドライバがGDIで分離、標準化されたために、アプリケーションの負荷は随分楽になった。プリンタに固有の制御はプリンタドライバに閉じ込めた。このため、プリンタメーカーが提供するプリンタドライバをホストコンピュータ内にインストールすれば、どんなアプリからでも印刷が可能になった。

【0443】また、Windows3.0では、プリンタのポートとして、RS-232Cやセントロニクスだけでなく、ネットワークも標準でサポートしている。

【0444】ネットワークプリンタのプロトコルは、図63に示すように構成されている。

【0445】図63は、図2に示した第2のステーションST2におけるネットワークプリンタのプロトコルを説明する図である。

【0446】図において、アプリケーション1560（図2に示したアプリケーションプログラム66に対応する）は自分自身が管理するメモリエリアではなく、上記ステーションST1（Macintosh）と同様にGDIモジュールに対して描画する。GDIモジュールはGDI関数のメタファイル1562を作ってプリンタドライバ1563に制御を渡す。プリンタドライバ1563は、1個1個のGDI関数をCaPSLコードに変換して、一時ファイル1565を作り、プリントマネージャー1566に制御を渡す。プリントマネージャー1566（図2に示したプリンタマネージャー65に対応する）は、通常は一時ファイル1565をシリアルポートからローカルプリンタに出力するが、本実施例においては、SPクライアント1569（図2に示した通信プログラム63に対応する）を介して、指定されたネットワークプリンタ（カラーSPサーバー）に伝送する。

【0447】ここでは、SPクライアント1569、LANマネージャー1568と同じレベルのネットワークドライバとして機能する。1572はプロトコル（TCP/IPプログラム62）、1573はイーサネットプロトコル、1574はイーサネットボード、1610はSPサーバー上のインタプリタ（図2に示したPDLインタープリタープログラム84に対応する）、1611はイーサネットボード、1612はイーサネットプロトコル、1613はプロトコル（図2に示したTCP/I

Pプログラム82に対応する）、1617はNetBIOS用のポート、1614はSocket用のポート、1615はSPサーバー（図2に示した通信プログラム83に対応する）である。

【0448】また、Windows準拠のアプリケーション1560は、GDIモジュール（描画の関数群）1561をコールしながら1ページの文章や絵を書いて行く。これを受けてGDIモジュール1561は、GDIのメタファイル1562に変換してファイルにセーブする。ページの終りで、GDIプリンタドライバをロードし、起動し、メタファイル(MF)1562のファイル名をドライバに引き渡す。プリンタドライバ1563では、プリンタに固有の処理を行う。例えば、メタファイル-CaPSLコード(MF-CaPSL)変換を行う。その結果、CaPSLファイルがセーブされる。プリンタドライバ1563は再び、GDIモジュール1564に対して印刷の実行を指示する。これを受けて、GDIモジュール1564は以下(1)～(3)のいずれかの処理をする。

【0449】(1) PrintManager（スプーラ）を起動し、その後の制御を任せ、プリンタドライバに対して印刷（スプール）の完了を返す。

【0450】(2) PrintManagerを起動せず、直接ローカルプリンタかネットワークプリンタにCaPSLファイルを転送する。（その間アプリケーションは作業を中断）

(3) 仮想プリンタ（ファイル）に印刷する。

【0451】このように構成された第1のサーバー装置において、プリンタドライバ1563が各ホスト特有のプリンタインタフェース（本実施例ではプリンタインタフェースGDI）に基づいて変換した印刷画像情報をサーバー装置に接続される画像処理装置（スキャナプリンタ）のプリント言語（本実施例ではCaPSL）に基づく画像出力情報に変換すると、このプリントドライバ1563の下位レイヤとしてのSPクライアント1569とSPサーバー1615とがネットワーク96を介して画像処理装置のプリント言語に基づく画像出力情報を通信転送させることにより、各ホストコンピュータ特有のプリンタインタフェースに基づいて変換した印刷画像情報を、サーバー装置に接続されるいかなる言語仕様に基づく画像処理装置でも市販のアプリケーションプログラムから印刷処理を高速に可能とする。なお、本実施例ではクライアント通信制御手段はSPクライアント1569に対応し、サーバー通信制御手段はSPサーバー1615に対応する。

【0452】なお、ネットワークプリンタに印刷する時は、PrintManagerを使わない方が良い場合がある。

【0453】また、本実施例ではTCP/IPのサービスを利用するためのポートとして、NetBIOS用のポート1571、Socket用のポート1570が用意されている。ネッ

トワークサービスプロトコル (NFS, FTP, LAN Manager, クライアント/サーバープロトコル等) は、これらのポート1571, 1570の上に構築される。なお、UNIXマシンにおいては、UNIXの4.3BSDと互換性を持って通信するために、Socket用のポート1570を使用する。

【0454】一方、マイクロソフト社製のネットワークサービスプロトコル (LANManager) を使用する場合は、マイクロソフト社が開発したNetBIOS用のポート1571を使用する。これは、Socketとは互換性が無いので、UNIXの側にLANManaher/Xと言うプロトコルを載せる必要があるためである。

【0455】図64は、図2に示した第2のステーションST2による印刷プロセスの一例を示すシーケンス図である。なお、標準のWindowsドライバは、DLLとして書かれている。

【0456】図65は、図63に示したSPクライアント1569のプログラムの構造を示す図であり、GDIインタフェース1569A, SPクライアントプログラム1569B, Socketコールルーチン1569C, 初期設定

```

<:Windows>;
<:Server>;
NetOpenJob() ----->
<----- ファイル名のハンドル
NetCloseJob() ----->
送信開始----->

```

以下、図66を参照しながら、図2に示したステーションST2 (IBM-PC/AT) と指定されたSPサーバーSP1との通信制御動作について説明する。

【0460】図66は、図2に示した第2のステーションST2とSPサーバーとのネットワーク接続状態を説明する図である。

【0461】図において、1620はアプリケーション (図2に示したアプリケーションプログラム66に対応する)、1621はGDI、1622はCaPSL-IVプリンタドライバ、1623はS/Pクライアント (図2に示した通信プログラム63に対応する)、1624はプロトコル (図2に示したTCP/IPプログラム62に対応する)、1625はイーサネットボードで、LAN96を介して指定されたSPサーバー装置と通信し、ネットワークプリンタとして機能する指定されたプリンタ1630での印刷ジョブを要求する。1631がビットマップ画像で、プリンタ1630に出力される。1632はCaPSLラスタイメージプロセッサ (図2に示したPDLインタープリタープログラム84に対応する) で、イーサネットプロトコル1635、プロトコル1634 (図2に示したTCP/IPプログラム82に対応する)、SPサーバー1633 (図2に示した通信プログラム83に対応する) を介してCaPSLコードをラスタイメージに変換する。

ダイヤログ1569D等から構成される。

【0457】なお、SPクライアント1569のプロトコル1572は、Windows 3.0においては、ネットワークプリンタドライバとして機能し、そのプロトコルの特徴は、TCP/IPを用いた事によって、必然的に、クライアント/サーバー型のプロトコルになることであり、Point-to-Pointの通信を行うことを意味する。

【0458】一方、NetWareやLANManagerは、ネットワーク分散型のプロトコル、例えばLANManagerの場合、ファイル名のハンドルを、ネットワーク側からもらって、そこにデータを書き込むので、ローカルにスプールファイルを作る必要が無い。これに対して、本実施例に示すSPクライアント1569のプロトコル1572では、TCP/IPがネットワークOSをサポートしていないため、クライアント自身がファイル名のハンドルを返し、その結果、ローカルなスプールファイルが下記のように作られる。

【0459】マクロのシーケンスは、例えば以下の通り：

```
<:Client>;
```

【0462】この図に示すように第2のステーションST2には、CaPSL-IVプリンタドライバ1622、SPクライアント1623がインストールされており、SPサーバー側にはSPサーバー1633がインストールされている。

【0463】プリンタドライバ1622は、IBM-PC/ATのアプリケーションからTCP/IPプロトコル1624を介してネットワーク96上のスキャナープリンタサーバー (SPサーバー) に印刷する為の、スキャナープリンタドライバとして機能する。

【0464】具体的には、下記 (1), (2) の処理を行う。

【0465】 (1) プリント時にGDI描画サブルーチンがコールされた時、プリンタドライバ1622がそれと等価なCaPSLコードを生成する。

【0466】 (2) 生成したCaPSLコードを、SPクライアント1623がドライバを通じてSPサーバー1633に伝送する。SPクライアント1623は、SPサーバー1633に接続されているプリンタ1630にCaPSLコードを伝送するための通信制御プログラムであり、TCP/IPプロトコル1624、ネットワーク96を介して通信する。SPクライアント1623の基本機能は以下の通りである。

【0467】 (1) TCP/IPプロトコル1624を介して

SPサーバー1633とEnd-to-Endのリンクを張る。

【0468】(2) CaPSL-IVプリンタドライバ1622から受け取ったCaPSLデータを、SPサーバー1633に送る。

【0469】(3) アプリケーションプログラム68を用いてSPサーバー1633に原稿のスキャンをさせる場合、それを受信し、スキャナアプリケーションプログラム68に送り返す。なお、SPサーバー1633は、SPサーバー装置上でデーモンとして常に走っており、クライアントからの受信を待っている。SPサーバー1633の基本機能は以下の通りである。

【0470】(1) SPクライアント1623から受け取ったCaPSLデータを、CaPSLインタープリタ(図2に示したPDLインタープリタープログラム84に対応する)に渡す。

【0471】(2) 原稿スキャンプログラムを起動し、受け取ったデータをSPクライアント1623に送る。

【0472】以下、図65に示した初期設定ダイアログ1503Dの構成および作成方法について説明する。

【0473】一般のアプリケーションプログラムは、印刷に関して余り多くのパラメータを制御していない。特に、アプリケーションプログラムがWINDOWSのGDIインタフェースに伝えるパラメータは、下記の通りである。

【0474】(1) GDIで描画する時のページバッファメモリ領域に関するパラメータ

(2) 描画データに対応したGDI関数に関するパラメータ

(3) 印刷開始コマンドに関するパラメータ

この結果、もし、GDI-CaPSL変換ドライバ(変換プログラム64)が選択されている場合、CaPSLのメタファイル1562が作成される。これを受けて、WINDOWSのプリントマネージャーがプリントドライバに対してダイアログボックスを開く用に要求し、オペレータに対して、以下のようなパラメータ(1)～(3)を入力する用に催促する。

【0475】(1) ページ毎の印刷枚数(デフォルト値は1枚)

(2) 印刷するページ数(デフォルト値は全ページ)

(3) 印刷順序(デフォルト値は逆順(印刷順序が最終ページから先頭ページ))

そして、WINDOWSのプリントマネージャー1566は、SPクライアント1569に対して、CaPSLコードのメタファイル1562を指定されたローカルSPプリンタに伝達するように要求する。これを受けて、SPクライアント1569は、ローカルプリンタにCaPSLコードを送る代わりに、ネットワーク(LAN96)を介してS/Pサーバー1615にCaPSLコードを送信する。

【0476】なお、印刷に必要なパラメータの入力方法

は、プリンタドライバに依存するが、印刷時にその都度セットするのは煩わしいパラメータは、WINDOWSの「印刷」メニューとは別に、「用紙設定」のメニューから随時設定できるように構成されている場合が多い。この用紙設定メニューで入力するパラメータとしては、用紙サイズ(A4/A3)、紙送り方向(ポートレート/ランドスケープ)等であり、プリンタがローカルプリンタであれば、これらのパラメータで十分であるが、ネットワークプリンタに出力する場合には、SPクライアント1569としては、さらに、SPサーバーのIPアドレス、ホストコンピュータ名、プリンタの種類(カラー/モノクロ複写装置、バブルジェット(BJ)カラー複写装置の選択)が必要となる。

【0477】そこで、本実施例における初期設定ダイアログ1569Dでは、下記のパラメータ設定を要求する。

【0478】(1) SPサーバーのIPアドレス(例えば192.9.200.101)

(2) クラス (B)

(3) サブネットマスク (255.255.255.0)

(4) デフォルトプリンタの種類(CLC/BJ)

(5) 通信プロトコル (TCP/IP)

これらのパラメータは、設定後、特定のファイルにセーブされなければならない。また、SPクライアントは、起動時にこのファイルを参照し、必要なパラメータをセットする。なお、初期設定ダイアログ1569Dには、WINDOWSの様式に従い、「OK」(設定終了、ファイル更新状態を指示する)ボタン、「Cancel」(設定値は前の状態に戻し、ファイルは更新しない状態を指示する)ボタン、「Option」(2枚目のダイアログを開く必要がある状態を指示する)ボタンを備えている。

【0479】以下、図67を参照しながら、図2に示した第2のステーションST2上のWINDOWSと各ドライバとの関係について説明する。

【0480】図67は、図2に示した第2のステーションST2上のウィンドウズと各ドライバとの関係を説明する図である。

【0481】この図において、DTPアプリケーション1640(図2に示したアプリケーションプログラム66に対応する)から印刷しようとする、プリントマネージャー(Printing Manager)1645(図2に示したプリントマネージャー65に対応する)がコールされる。プリントマネージャー1645は、プリンタドライバ1643に対し、描画メモリ領域を確保させそのポインタを提供させる。プリントマネージャー1645は、そのポインタをアプリケーション1640に知らせて、そこにGDIフォーマットで描画させる。従って、アプリケーション1640が描画を完了すると、そのデータ

はすっかりプリンタドライバ1643に渡されている事になる。プリンタドライバ1643は、そのGDI関数を1個1個、順番にCaPSLコードに変換してメモリに蓄え、S/Pクライアント(S/PCient)1646

(図2に示した通信プログラム63に対応する)にそれをS/Pサーバー1602(図2に示した通信プログラム83に対応する)側に伝送する様に依頼をする。S/Pクライアント1646(図2に示した通信プログラム63に対応する)とS/Pサーバー1602との間は、データ転送プロトコルによって結ばれており、プリンタドライバ1643から渡された、変換済みのCaPSLコードをS/Pサーバー1602に転送する。

【0482】なお、1650は通信プロトコル(図2に示したTCP/IPプログラム62に対応する)、1647は圧縮(ADCT)制御関数、1648は圧縮ボードである。1642はGDI-CaPSL変換ルーチン(図2に示した変換プログラム64に対応する)で、アプリケーションの出力GDIコールのすべての描画関数をCaPSLコードに置き換える。なお、ここで、使用するCaPSLは、CaPSL-IVをADCT拡張したものである。

【0483】この図において、プリンタドライバ1643は、GDI-CaPSL変換ルーチン1642の部分で変換されたデータの授受に関する制御を行う。具体的には、アプリケーションプログラムがEscape(STARTDOC)やEscape(ENDDOC)等の関数を発行した時に、プリントマネージャー1652から要求されるパラメータを返したり、SPクライアント1646のドライバのコマンドを制御する。プリンタドライバ1643は、ドラフトモード(Spoolなし)で動作する。プリンタドライバ1643は、GDI1641によって起動され、プリントマネージャー1652に必要なコマンドを発行する。プリンタドライバ1643は、後述する図68に示すGDIモジュール1644に処理結果を通知する。

【0484】図68は、図2に示した第2のステーションST2上とSPサーバーとの各ドライバとの関係を説明する図であり、図67と同一のものには同じ符号を付してある。

【0485】この図において、SPクライアント1647のドライバはプリントマネージャー1652から要求される各種のハレベルデバイスマネージャールーチン(DriverOpen, DriverClose, Control, FSRead, FSWrite, Status, KillIO)を提供する。

【0486】以下、図69、図70を参照しながら各ホストコンピュータのプリンタインタフェースにより変換された画像情報のSPサーバーへの通信処理について説明する。

【0487】図69は本発明に係る各ホストコンピュー

タのネットワークプリンタ処理プロセスを説明する概念図である。

【0488】この図に示すように、ホストコンピュータHOSTAは、ポストスクリプト対応のアプリケーション5000からプリドライバ5002を介してネットワーク5003上のポストスクリプト対応のプリンタ5004に印刷する場合に対応し、アプリケーションからプリンタ5004までがすべてポストスクリプトで統一されている。この場合、アプリケーション5000とプリドライバ5002とがOS5001を介在してプリントデータが処理されないため、ポストスクリプト対応のプリンタ以外はネットワーク5003に接続することができない。

【0489】一方、ホストコンピュータHOSTBは、クイックドロース対応のアプリケーション5010からOS5001、プリンタドライバ5011を介してネットワーク5003上のポストスクリプト対応のプリンタ5012に印刷する場合に対応し、特にポストスクリプトのエミュレーション機能(プリンタ5012内にエミュレータを設ける)を用いてポストスクリプト対応のプリンタ5012がクイックドロータイプのプリンタデータを処理する場合に対応する。

【0490】一方、ホストコンピュータHOSTCは、クイックドロース対応のアプリケーション5020からOS5001を介して、プリンタドライバ5021上でクイックドロースを特定のプリンタの言語に変換して、ネットワーク5003上のSPサーバー5022を介してSPサーバー5022がドライブする特定のプリンタ5023(本実施例ではスキャナプリンタとして、カラーレーザ複写装置(CLC)、バブルジェットカラー複写装置(BJC)を採用している)に特定のプリンタ言語に従うプリントデータを処理させる場合に対応する。なお、プリンタ言語としては、CaPSL(商品名)、LIPS(商品名)、PCL(商品名)等がある。これにより、各プリンタ言語に対応させてプリンタドライバ5021を構成すれば、どのようなプリンタであってもネットワークプリンタとしてシステムを構築させることができ、プリンタ選択の余地が大幅に広がる。すなわち、印刷目的に応じたプリンタ選択が可能となり、カラー印刷の別、プリンタ処理速度、印刷コストに対応した最適なプリンタをネットワークプリンタとして機能させることができる。

【0491】図70は本発明に係るスキャナプリンタサーバーシステムのSPサーバと各ホストコンピュータのSPクライアントとの交信プロセスを説明する概念図であり、図68と同一のものには同じ符号を付してある。

【0492】例えば図68に示すSPクライアント1647は直接的にはTCPプロトコル1650Bにすべてのコマンドとデータを書き込むが、概念上は、ネットワーク96を介して接続されたSPサーバー1602と交信

しているように処理を行う。従って、End-to-Endのプロトコル（クライアントプロトコル2001-サーバープロトコル2004）が必要になる。この際、SPクライアント1647とSPサーバー1602とは、互いにパケットをやり取りすることにより、上位レイヤに通信サービスを提供する。なお、パケットは、サーバー/クライアントパケット、ドキュメント制御パケットに分かれている。

【0493】特に、SPクライアント通信プログラムは、ドライバインタフェース2002、クライアントプロトコル本体2001、TCP/IPインタフェース2003に分かれ、TCP/IPプロトコル1650とTCP/IPプロトコル1603とは厳密に標準に合致する必要があるが、TCP/IPインタフェース2003の部分は各ホストマシン（IBMホスト、Macホスト、SUNホスト、NEXTホスト）により異なる。

【0494】本実施例ではプリンタドライバの下位レイヤとしてクライアントプロトコル2001とサーバープロトコル2004とをネットワーク96を介して接続してSPサーバー装置に、例えばCaPSLに変換された画像情報を伝送して接続されるスキナプリンタとしてのカラーレーザ複写装置（CLC）、バブルジェットカラー複写装置（BJC）より印刷させている。

【0495】図71は本発明に係るスキナプリンタサーバーシステムによるパケット処理手順の一例を示す図である。

【0496】この図において、2500、2600、2700は独立したプログラムの処理単位（タスクC～A）で、例えばレイヤ毎に分割された通信プログラムのレイヤ毎のパケット処理ルーチンに対応する。なお、本実施例においては、各レイヤを従来のレイヤ概念と識別するためレベルと呼ぶ。また、説明上レベルが3つとすることが、これ以上であっても同様である。

【0497】この図に示されるように、プログラムの処理単位2700で示すタスクAがレベル3（LEVEL-3）に、プログラムの処理単位2600で示すタスクBがレベル（2LEVEL-2）に、プログラムの処理単位2500で示すタスクCがレベル1（LEVEL-1）に対応する。

【0498】特に、レベル1は、イーサネットを介して外部インタフェース2900と直接データをやり取りするためのプロトコルで、主にネットワークにまつわる仕事のやり取りを行う。レベル2は、中間レベルのプロトコルで、主に端末とネットワークとの間で信頼性のあるデータのやり取りを行う。レベル3は、上位レベルのプロトコルで、ネットワークを介してEnd-to-Endで信頼性のある通信路を確保する。なお、実際のアプリケーションはレベル3の上に構築されるものとする。

【0499】このように本発明に係る第5のサーバー装置の通信処理方法によれば、低位の通信プログラムから

高位の通信プログラムを連続的に実行してバッファメモリに画像情報を記憶させる際に、低位の通信プログラムから上位の通信プログラムに画像情報に対するバッファメモリへの展開アドレスを催促し、該催促に従って高位の通信プログラムから低位の通信プログラムに伝達される展開開始アドレスに基づいて低位の通信プログラムによりバッファメモリに画像情報を展開させるので、低位の通信プログラムから高位の通信プログラムを連続的に実行してバッファメモリに画像情報を転送して記憶させる際の各通信プログラム間のオーバーヘッドを抑えることが可能となる。

【0500】具体的には、まず、タスクC（レベル1）が外部インタフェース2900からデータを受信使用とする時、タスクCは実際の受信バッファを確保するとことなく、その代わりにタスクB（レベル2）に対して実メモリのポインタ（アドレス）を要求する。タスクCはタスクBから受け取ったポインタを開始アドレスとする所定メモリエリア（仮想メモリ）に対して受信データを書込む。タスクBがタスクCに引き渡すポインタ2602は、タスクBが実際に持つメモリのポインタでなくとも良い。本実施例では、タスクAのみが実メモリ2701を確保し、タスクBとタスクCは実メモリ2701へのポインタ2702、2602を知るのみである。このように、タスクBはタスクAから受け取ったポインタ2702をポインタ2602として引き渡している。この場合、タスクCが受信したデータ2800は直接タスクAの実メモリ2701に書込まれる。タスクB、Cも、そのデータの中身を直接参照することができる。このように構成することにより、1回のデータ受信に関して内部でのデータのリード/ライトは1回で済み、最小のデータ伝送回数で処理できる。この結果、タスク毎に受信バッファメモリを設ける場合に比べて、格段に処理速度（パケット通信処理速度）が向上する。なお、本実施例では高速化の障害となるような層構造を持たず、レベルというプログラム処理単位でパケット通信処理を実行する。また、詳細は省略するが、パケットは各レベル1～3に共通のヘッダ、例えばprotocol, channel, packet type, packet id, modifier, pk size [2]の8バイトを備える8バイト構成となつて定義されている。このうち、3つのレベルを分離するのは、「packet type」の上位4ビットだけで行っており、その他の要素はすべてのレベルで共通に使用される。各レベルは、自分のレベル宛に送られたパケットのみを処理する。ただし、他のレベルのパケットは、呼ぶことは自由であるが、書込むことは禁止される。

【0501】これに対して、従来もしくは通常のプロトコルでは、上位レイヤのパケットのすべては、下位レイヤのデータパケットによって伝送されるが、例えばレベル1のパケットは、レベル1～レベル3の各ヘッダを持

つ必要があり、伝送効率が低下するばかりでなく、ヘッダの解析ルーチンを持つ必要も生じ、メモリ資源の無駄と処理時間の無駄が発生する。従って、本実施例のように、「packet type」が自分より上位のタイプであれば、すべて上位レベルに伝えることにより、各レベルでのヘッダ解析処理負担を軽減することができる。なお、上記実施例ではデータの受信処理を例にして説明したが、送信の場合にも同様に本発明を適用できることは言うまでもない。

【0502】以下、図72に示すフローチャートを参照しながら本発明に係るスキャナプリンタサーバーシステムに接続されるプリンタにおけるカラー／白黒印刷制御動作について説明する。

【0503】図72は本発明に係るスキャナプリンタサーバーシステムに接続されるプリンタにおけるカラー／白黒印刷制御手順の一例を示すフローチャートであり、図2に示すシステム全体制御プログラム93によって実行される。なお、(1)～(11)は各ステップを示す。

【0504】先ず、ステップ(1)において、通信プログラム83を介して入力された画像情報からアプリケーションプログラムの種別がカラー／白黒のいずれかを判定し(1)、カラーアプリケーションプログラムである場合には、ステップ(2)において、カラーアプリケーションプログラムにおいて、印刷指定がカラー印刷かそれとも白黒印刷かを判定し(2)、カラー印刷の場合には、ステップ(3)において、プリンタがカラープリンタかそれとも白黒プリンタかを判別し(3)、カラープリンタ(例えばカラーレーザ複写装置(CLC))である場合には、ステップ(6)において、カラーアプリケーションプログラムによるカラー印刷情報をカラープリンタでカラー印刷するようにプリンタを制御する。

【0505】一方、ステップ(3)の判定で白黒プリンタであると判定された場合は、ステップ(7)において、カラーアプリケーションプログラムによるカラー印刷情報を白黒プリンタで白黒印刷するようにプリンタを制御する。

【0506】一方、ステップ(2)の判定で白黒情報であると判定された場合は、プリンタがカラープリンタ指定かそれとも白黒プリンタ指定かを判別し(4)、カラープリンタ(例えばカラーレーザ複写装置(CLC))である場合には、ステップ(8)において、カラーアプリケーションプログラムによる白黒印刷情報をカラープリンタで白黒印刷するようにプリンタを制御する。

【0507】一方、ステップ(4)の判定で白黒プリンタであると判定された場合は、ステップ(9)において、カラーアプリケーションプログラムによる白黒印刷情報を白黒プリンタで白黒印刷するようにプリンタを制御する。

【0508】一方、(1)の判定で白黒アプリケーションプログラムである場合は、プリンタがカラープリンタ指

定かそれとも白黒プリンタ指定かを判別し(5)、カラープリンタ(例えばカラーレーザ複写装置(CLC))である場合には、ステップ(10)において、白黒アプリケーションプログラムによる白黒印刷情報をカラープリンタで白黒印刷するようにプリンタを制御する。

【0509】一方、ステップ(5)の判定で白黒プリンタであると判定された場合は、ステップ(11)において、白黒アプリケーションプログラムによる白黒印刷情報を白黒プリンタで白黒印刷するようにプリンタを制御する。

【0510】このように、最終的に白黒プリントが行われる場合には、カラーアプリケーションプログラムによる印刷設定がカラーでプリンタが白黒プリンタ指定の時、カラーアプリケーションプログラムにより印刷設定が白黒で、プリンタがカラープリンタの時、カラーアプリケーションプログラムによる印刷設定が白黒で、プリンタが白黒プリンタの時、白黒アプリによりプリンタがカラープリンタの時、白黒アプリによりプリンタが白黒プリンタの時である。

【0511】上記分類の中で、正常なプリントはカラーアプリケーションプログラムによる印刷設定が白黒で、プリンタが白黒プリンタの時、白黒アプリケーションプログラムによりプリンタが白黒プリンタの時であり、その他の場合は、アプリケーションプログラムの印刷設定と、SPサーバのプリンタハードウェアにおいて、カラー、白黒設定が整合していないため例外処理として処理する。

【0512】なお、ネットワークプリンタとして接続可能なプリンタの種類は、カラープリンタとして、上記カラーレーザ複写装置(CLC)、BJカラー複写装置(BJC)、市販カラープリンタであり、白黒プリンタとしては、市販のページプリンタ、市販ラインプリンタ等である。

【0513】以下、図73を参照しながら本発明に係るスキャナプリンタサーバーシステムにおける印刷ジョブの流れについて説明する。

【0514】図73は本発明に係るスキャナプリンタサーバーシステムに接続される各入出力機器における印刷ジョブの流れを説明する図である。

【0515】この図に示すように、ホストマシンのオペレータがアプリケーションプログラムにデータ入力を行い、さらにホストコンピュータ上でプリンタ指定を行って、プリント出力の指示等を行うと、アプリケーションプログラムはPDL変換ソフト3001(例えば変換プログラム54)に出力データ、プリンタ指定情報等を渡す。これを受けて、PDL変換ソフト3001は、アプリケーションプログラムから受け取ったデータに対して、プリンタ指定情報等に基づきPDL変換を行い、PDL変換したデータをSPサーバとのSPクライアントソフト3002(例えば通信プログラム53)に渡す。これを受けて、SPクライアントソフト3002は、S

Pサーバーのサーバソフト3003（通信プログラム83）とネットワーク（図示しないイーサネット等）を介して通信を行い、PDL変換されたデータをSPサーバーソフト3003に送る。これを受けて、SPサーバーソフト3003は、SPクライアントソフト3002からデータを受取り、全体制御ソフト3004（システム全体制御プログラム93）にデータが送られてきたことを伝える。これを受けて、全体制御ソフト3004は、送られてきたデータと、SPサーバーの状態に基づき、例えばPDLインタープリタープログラム84等のプリント処理ソフト3005にデータの処理を依頼する。これを受けて、プリント処理ソフト3005は依頼されたデータを画像入力装置制御ソフト3007（デバイスドライバ86）が受け入れられる形に処理を行い、バンドメモリ3006に結果を保存する。プリント処理ソフト3005の処理が終了すると、終了を全体制御ソフト3004に伝える。全体制御ソフト3004は、画像入力装置制御ソフト3007にプリント出力を依頼する。これを受けて、画像入出力装置制御ソフト3007は、バンドメモリ3006からデータを読み取り、画像入出力装置インターフェース3008にデータを送る。これを受けて画像入出力装置インターフェース3008は、画像入出力装置制御ソフト3007から送られてきたデータをスキナプリンタ、例えば指定されたS/Pサーバ装置に接続されるカラーレーザ複写装置のプリンタ部に出力する。以上の処理により、ネットワーク上のホストコンピュータのアプリケーションプログラム（ネットワーク上のプリンタのページ記述言語と異なるページ記述言語のアプリケーションプログラム）からネットワーク上の指定したプリンタに所望のデータをプリントアウトすることができる。

【0516】なお、白黒プリントが行なわれる場合、最終的に白黒で出力されるわけであるが、アプリケーションプログラムの設定、SPサーバーの設定等の違いをどの処理で認識し、吸収するかということが問題になるが、本実施例ではプリントアウトする全体の処理流れの中で、上記カラー／白黒印刷処理の設定を吸収できる箇所として、ホスト側のアプリケーションプログラムからの出力データを処理するPDL変換ソフト3001、SPサーバのプリント処理ソフト3005、SPサーバの画像入出力装置制御ソフト3007で実行可能である。

【0517】上述したように、白黒／カラー印刷処理での白黒印刷は、1. カラーアプリケーションプログラムによる印刷設定がカラーでプリンタが白黒プリンタの時、2. カラーアプリケーションプログラムにより印刷設定が白黒で、プリンタがカラープリンタの時、3. カラーアプリケーションプログラムによる印刷設定が白黒で、プリンタが白黒プリンタの時、4. 白黒アプリケーションプログラムによりプリンタがカラープリンタの時、5. 白黒アプリケーションプログラムによりプリンタが白黒プリン

タの時であるので、場合に分けて説明する。

【0518】先ず、1. カラーアプリケーションプログラムによる印刷設定がカラーでプリンタが白黒プリンタの時は、オペレータのホストコンピュータ上のダイアログに表示する警告例は「カラー印刷設定ですが、指定したプリンタは白黒プリンタです。このプリンタで印刷しますか？」となる。この場合、アプリケーションプログラムからのデータはカラーデータとなる。プリンタが白黒のため、カラー／白黒変換を行う必要がある。カラー／白黒の吸収はPDL変換ソフト3001、プリント処理ソフト3005で行うことが可能である。なお、画像入出力装置制御ソフト部3007では白黒プリンタのつくりのため、通常はカラー／白黒変換は出来ないが、白黒プリンタ用でも、カラー入力可能な場合は別である。

【0519】そこで、PDL変換ソフト3001で吸収する場合は、PDL変換ソフト3001は白黒プリンタ指定を認識し、PDL変換を行なう際にカラーデータを白黒データに変換する。

【0520】また、プリント処理ソフト3005で吸収する場合は、カラーデータが送られてくると、全体制御ソフト3004の指示により、バンドメモリ3006に白黒展開をおこなう。

【0521】さらに、画像入出力装置制御ソフト3007で吸収する場合は、自分自身は、白黒プリンタのソフトであるから、全体制御ソフト3004から、バンドメモリ3006にはカラーデータがあることを伝えられなければならない。そして、画像入出力装置インターフェース3008にデータ転送を行うときに、カラーデータを白黒に変換する。

【0522】一方、2. カラーアプリケーションプログラムにより印刷設定が白黒で、プリンタがカラープリンタの時は、オペレータのホストコンピュータ上のダイアログに表示する警告例は「白黒印刷設定ですが、指定したプリンタはカラープリンタです。このプリンタで印刷しますか？」等となる。この場合、アプリケーションプログラムからのデータは白黒データとなる。プリンタがカラーのため、白黒／カラー変換を行う必要がある。白黒／カラーの吸収はPDL変換ソフト3001、プリント処理ソフト3005で行うことが可能である。画像入出力装置制御ソフト3007ではカラープリンタのつくりのため、通常は白黒／カラー変換は出来ない。カラープリンタ用でも、白黒入力可能なつくりにしていれば、可能である。

【0523】そこで、PDL変換ソフト3001で吸収する場合は、PDL変換ソフト3001はカラープリンタ指定を認識し、PDL変換の際に白黒データをカラーデータに変換する。

【0524】また、プリント処理ソフト3005で吸収する場合は、白黒データが送られてくると、全体制御ソフト3004の指示により、バンドメモリ3006にカ

ラー展開を行う。

【0525】さらに、画像入出力装置制御ソフト3007で吸収する場合は、自分自身は、カラープリンタのソフトであるから、全体制御ソフト3004から、バンドメモリ3006には白黒データがあることを伝えられなければならない。画像入出力装置インタフェース3008にデータ転送を行うときに、白黒データをカラーに変換する。

【0526】一方、3. カラーアプリケーションプログラムによる印刷設定が白黒で、プリンタが白黒プリンタの時は、警告なし（正常）で、PDL変換部3001は初めから、白黒モードで変換を行う。プリント処理ソフト3005も白黒モードで処理を行う。画像入出力装置制御ソフト3007は通常通りの動作を行う。

【0527】一方、4. 白黒アプリケーションプログラムによりプリンタがカラープリンタの時は、オペレータのホストコンピュータ上のダイアログに表示する警告例は「指定したプリンタはカラープリンタです。このプリンタで印刷しますか？」等となる。

【0528】この場合は、以下、上記2.の場合と同様である。

【0529】一方、5. 白黒アプリケーションプログラムによりプリンタが白黒プリンタの時は、警告なし（正常）で、以下、上記3.の場合と同様である。

【0530】以上の実施例に関して、吸収する際の条件を図74に示す。

【0531】図74は、図73に示した各入出力機器の白黒印刷吸収条件を示す図である。

【0532】この図において、○は上記1.～5.の処理実行箇所を示し、a～cは処理開始位置の条件を示し、印刷条件の指定をホストコンピュータ側に近い段階で行う（a条件）場合、印刷条件の指定をS/Pサーバー側で行う（b条件）場合、印刷条件の指定を全体のデータの流れとして効率の良いところで行う（c条件）場合とがある。

【0533】このような条件から各処理を吸収する箇所は以下のように分類できる。

【0534】すなわち、条件aを満たさせる場合には、カラー（指定）／白黒（プリンタ）、白黒（指定）／カラー（プリンタ）の変換は、ホストコンピュータ側のPDL変換ソフト3001で全て行なうことになる。この場合、S/Pサーバー側は特別な処理を必要としない。

【0535】また、条件bを満たさせる場合には、プリント処理ソフト3005が全てのカラー（指定）／白黒（プリンタ）、白黒（指定）／カラー（プリンタ）の変換を行なうか、画像入出力装置制御ソフト3007が全てのカラー（指定）／白黒（プリンタ）、白黒（指定）／カラー（プリンタ）の変換を行なうか、プリント処理ソフト3005がカラー（指定）／白黒（プリンタ）変換を行ない、画像入出力装置制御ソフト3007が白黒

（指定）／カラー（プリンタ）の変換を行なうか、プリント処理ソフト3005が白黒（指定）／カラー（プリンタ）の変換を行ない、画像入出力装置制御ソフト3007が白黒（プリンタ）変換を行なうかという場合がある。

【0536】さらに、条件cを満たさせる場合には、この場合、プリンタがカラープリンタの時、つまり、上記2.のようにカラーアプリケーションプログラムにより印刷設定が白黒で、プリンタがカラープリンタの時、4.のように白黒アプリケーションプログラムによりプリンタがカラープリンタの時は、プリント処理ソフト3005で白黒／カラー変換を行なう。

【0537】また、プリンタが白黒のとき、つまり、1.のようにカラーアプリケーションプログラムにより、印刷設定がカラーで、プリンタが白黒プリンタの時、3.のようにカラーアプリケーションプログラムにより印刷設定が白黒で、プリンタが白黒プリンタの時、5.のように白黒アプリケーションプログラムによりプリンタが白黒プリンタの時で、変換の必要のある時、条件aを満たさせる場合には、PDL変換ソフト3001でカラー／白黒変換を行なう。この変換の場合、通信データは全て白黒PDL変換済みデータとなる。以上が標準アプリケーションプログラムから白黒プリントを行う場合の説明である。

【0538】以下、図75～図77を参照しながら図2に示した指定されたSPサーバー装置SP1とステーションST3（UNIXマシン）とのネットワーク処理について説明する。

【0539】図75は、図2に示したSPサーバー装置SP1と第3のステーションST3とのネットワークシステムの概要を説明するブロック図である。

【0540】図において、4001はUNIXマシンで構成されるワークステーション（図2に示したステーションST3に対応する）、4002はSPサーバー（図2に示した通信プログラム83に対応する）で、ネットワークとしてのイーサネット4004（図2に示したイーサネットLAN96）を介してワークステーション4001と交信し、ワークステーション4001から要求されたプリントジョブを処理してネットワークプリンタとして指定されたプリンタ4003から印刷出力を制御する。

【0541】図76は、図75に示したワークステーション4001の機能を説明するブロック図である。

【0542】図において、4005はUNIXマシンで標準のプリントコマンド（lpr）、4006はUNIXマシンで標準のプリント用デーモン（常駐プログラム）としてのlpdプログラムであり、図2に示したlpr/lpdプログラム90に相当する。4007はワークステーション4001の外部記憶装置内に格納されているプリンタ管理用ファイル（/etc/print

cap)で、SPサーバー4002に接続されているプリンタ4003のエントリが登録されている。4008はプリントするファイルの内容の複製(データファイル)とプリントのときの制御情報(コントロールファイル)を一時的に格納する外部記憶装置内のスプール領域、4009は外部記憶装置内に保存されているファイルである。

【0543】図77は、図75に示したSPサーバー4002の機能構成を説明するブロック図である。

【0544】図において、4010は前記SPサーバー4002内のプリント常駐プログラムであるサーバlpdプログラム、4011はSPサーバー内の全体制御ソフトウェア(図2に示したシステム全体制御プログラム93に対応する)、4012はSPネットワークサーバー内のSP管理用情報記憶部で、SPサーバー4002に接続されているプリンタ、スキャナの情報が登録されている。4013はSPサーバー4002内の主記憶装置、4014はCaPSLインタプリタ(図2に示したPDLインタプリタプログラム84に対応する)、4015はSPコントローラ(図2に示したデバイスドライバ86)である。

【0545】以下、UNIXマシンで標準のプリントコマンドlprを用いたSPサーバー4002に接続されたプリンタ4003への出力動作について説明する。

【0546】先ず、ワークステーション4001において、出力すべきファイル4009のファイル名とプリンタ管理用ファイル4007の中に登録されている出力先のプリンタ4003のプリンタ名を引数で指定してプリントコマンド(lpr)4005を起動する。プリントコマンド(lpr)4005は、引数で指定されたファイル4009の複製をデータファイルとしてスプール領域4008に作成する。また、プリント時の制御情報を格納したコントロールファイルをスプール領域4008に作成する。プリントコマンド(lpr)4005からlpd4006に対して引数で指定されたプリンタ名を送信する。lpd4006は、プリントコマンド(lpr)4005からプリンタ名を受信してそのプリンタ名がプリンタ管理用ファイル4007に登録されていることを確認した後、プリントコマンド(lpr)4005に対して応答を送信する。

【0547】プリントコマンド(lpr)4005には応答を受信して動作を終了する。そして、lpd4006は、SPサーバー4002と通信回線としてのイーサネット4004を介してリンクした後、SPサーバー4002上で動作しているlpd4010に対して、プリンタ管理用ファイル4007に登録されている前記プリンタ名を送信する。lpd4010がこのプリンタ名を受信すると、SP管理用情報記憶部4012に登録されているプリンタ名と照合し、その情報を得た後、lpd4006に対して応答を送信し、さらに全体制御ソフト

ウェア4011に対してプリンタオープン要求を送信する。

【0548】全体制御ソフトウェア4011は、プリンタに関して必要な設定を行った後に、lpd4010に対してプリンタオープン要求応答を送信する。lpd4010は、全体制御ソフトウェア4011からプリンタオープン要求応答を受信したら、lpd4006からの送信を待つ。lpd4006は、lpd4010からの応答を受信すると、スプール領域4008に作成したデータファイルのサイズと名前をlpd4010に対して送信する。lpd4010は、lpd4006からデータファイルのサイズと名前を受信したら、SPサーバー4002内の主記憶装置4013にデータファイルのサイズ以上のRAMディスクが獲得できるかどうかを調べる。RAMディスクが獲得できる場合は、データファイル用のRAMディスクを獲得した後、lpd4006に対して応答を送信する。RAMディスクを獲得できない場合は、SP管理用情報記憶部4012に登録されているプリンタ4003に関するデフォルトの設定値を得た後、lpd4006に対して応答を送信する。lpd4006は、lpd4010からの応答を受信すると、スプール領域4008に作成したデータファイルの内容を読み出し、順次lpd4010に対して送信する。データファイルをすべて送信した後に、送信終了をlpd4010に対して送信する。lpd4010は、lpd4006からデータファイルの内容を受信する。データファイル用のRAMディスクが獲得できている場合は、RAMディスク内にデータファイルを作成し、受信したデータファイルの内容を格納する。データファイル用のRAMディスクが獲得できていない場合は、受信したデータファイルの内容を順次全体制御ソフトウェア4011へ送信する。データファイルのサイズ分のデータを受信した後、送信終了したら、lpd4006に対して応答を送信する。データファイル用のRAMディスクが獲得できていない場合は、全体制御ソフトウェア4011に対してプリント終了を送信する。lpd4006は、lpd4010からの応答を受信すると、スプール領域4008に作成したコントロールファイルのサイズと名前をlpd4010に対して送信する。lpd4010は、lpd4006から前記コントロールファイルのサイズと前を受信したら、SPサーバー4002内の主記憶装置4013にコントロールファイルのサイズ以上のRAMディスクが獲得できるかどうかを調べる。RAMディスクが獲得できる場合は、コントロールファイル用のRAMディスクを獲得した後、lpd4006に対して応答を送信する。RAMディスクが獲得できない場合は、SP管理用情報記憶部4012に登録されているプリンタに関するデフォルトの設定値を得た後、lpd4006に対して応答を送信する。lpd4006は、lpd4010からの応答を受信すると、スプール領域4

008に作成したコントロールファイルの内容を読み出し、順次1pd4010に対して送信するとともに、コントロールファイルをすべて送信した後に送信終了を1pd4010に対して送信する。

【0549】1pd4010は、1pd4006からコントロールファイルの内容を受信する。コントロールファイル用のRAMディスクが獲得できている場合は、RAMディスク内にコントロールファイルを作成し、受信したコントロールファイルの内容を格納する。コントロールファイル用のRAMディスクが獲得できていない場合は、受信したコントロールファイルの内容を破棄する。コントロールファイルのサイズ分のデータを受信した後、送信終了を受信したら、1pd4006に対して応答を送信する。1pd4006は、1pd4010からの応答を受信すると、スプール領域4008に作成したデータファイルおよびコントロールファイルを削除し、SPサーバー4002とイーサネット4004を介して張っていたリンクを切断する。

【0550】1pd4010は、コントロールファイル用のRAMディスク内にコントロールファイルが作成されている場合は、その内容を読み出しプリンタに関する設定値を決定する。データファイル用のRAMディスク内にデータファイルが作成されている場合は、データファイルの内容を読み出し、順次全体制御ソフトウェア4011に対して送信する。データファイルの内容をすべて送信した後に、全体制御ソフトウェア4011に対してプリント終了を送信する。全体制御ソフトウェア4011は、1pd4010から受信した前記データファイルの内容を、CaPSLインタプリタ4014へ送信して、CaPSLに従ってイメージ展開を行った後、SPコントローラ4015へプリント要求を行う。また、SPコントローラ4015は、全体制御ソフトウェア4011からのプリント要求を受けて、プリンタ4003へ出力する。

【0551】図78は本発明に係るスキャナプリンタサーバーシステムと各ホストとのデータロード処理状態を説明するネットワーク図である。

【0552】図において、901はSPサーバーで、不揮発性メモリ901Aを有し、各種のリスト、プログラム等を記憶している。908はスキャナプリンタで、SPサーバー901と上述したインタフェースボードを介して接続されている。902、903はダウンロードホストコンピュータ（ダウンロードホスト）で、各ホストコンピュータに対応する、例えばハードディスク等で構成される記憶装置902A、903Aを備えている。905、906はクライアントホストで、ネットワーク907を介してSPサーバー901にプリント要求、スキャナ要求を出力する。

【0553】図79は、図78に示したSPサーバー901およびダウンロードホスト902、903の記憶情

報を説明するデータ構成図である。

【0554】この図に示されるように、SPサーバー901の不揮発性メモリ901Aには、あらかじめオペレータによって作成された特定の1台または優先度を付けた複数台のダウンロードホスト（本実施例ではダウンロードホスト902、903）の情報リスト（ダウンロードホストリスト）911、印刷時に必要なフィルタフォントの情報リスト（フィルタフォントリスト）912、ダウンロードプログラム913等が記憶されている。なお、あらかじめオペレータによって作成された特定の1台または優先度を付けた複数台のダウンロードホスト（本実施例ではダウンロードホスト902、903）の情報リスト（ダウンロードホストリスト）911および印刷時に必要なフィルタフォントの情報リスト（フィルタフォントリスト）912、ダウンロードプログラム913は、例えばテキストファイル形式で記憶されている。

【0555】一方、ダウンロードホスト902の記憶装置902Aには、SPサーバー901に必要な各種プログラム915（図2に示すSPサーバーSP1内のシステム全体制御プログラム93等の各種プログラム）、印刷を行う際に必要なフィルタおよびフォントデータ（フィルタフォントデータ）916、SPサーバー901の立ち上げに必要な初期設定パラメータファイル917、上記915～917をSPサーバー901にアップロードするためのアップロードプログラム918が記憶されている。

【0556】このように構成された本発明に係る第2のサーバー装置によれば、本体上に設けられた不揮発性メモリ901Aが特定の1台または優先度が付けられた複数台のホストコンピュータから所望の情報をダウンロードするためのホストコンピュータ先指定情報、前記所望の情報をダウンロードするためのダウンロードプログラム、プリント処理機能実行に必要なフォント情報、プリント処理機能実行に必要な印刷制御情報等を記憶することにより、特定の1台または優先度が付けられた複数台のホストコンピュータから所望の情報をダウンロードする際のホストコンピュータ先および印刷制御情報の変更処理を一元管理することを可能とする。

【0557】また、特定の1台または優先度が付けられた複数台のホストに記憶されたサーバー装置を立ち上げて初期設定するためのシステムプログラムおよび所望の情報をダウンロードすることにより、必要に応じてサーバー装置を立ち上げて初期設定するためのシステムプログラムおよび所望の情報を特定の1台または優先度が付けられた複数台のホストコンピュータ中の1つのホストコンピュータから適時ダウンロードすることを可能とする。

【0558】また、電源投入時に、ホストコンピュータ先指定情報に基づいて複数のホストコンピュータの動作

状態を確認して1つのホストコンピュータをマスタダウンロードホストコンピュータとして登録することにより、ネットワーク上のホストコンピュータ動作状態が変動しても、必要に応じてサーバー装置を立ち上げて初期設定するためのシステムプログラムおよび所望の情報を特定の1台または優先度が付けられた複数台のホストコンピュータ中の1つのホストコンピュータから適時ダウンロードすることを可能とする。

【0559】なお、SPサーバー901は後述するように電源立上げ時に、ネットワーク907上のダウンロードホストのチェックを行い、ダウンロードホストが複数設定されている場合は、動作中で最優先のものをマスタダウンロードホストとし、他のダウンロードホストをスレーブホストとし、ダウンロードホストが1台のみ設定されている場合には、そのホストコンピュータをマスタダウンロードホストとする。

【0560】さらに、ダウンロードホストのチェックを行う時に、いずれのダウンロードホストも動作していなかった場合、後述するようにSPサーバー901のエラー・ログ・ファイルに記録し、あらかじめ設定された最大待ち時間が経過するまで、あらかじめ設定された一定時間毎に再びダウンロードホストのチェックを行い、最大待ち時間が経過すると、立ち上げ失敗エラーとして処理を終了する。

【0561】また、ダウンロードホストに記憶されているSPサーバー901の各種プログラム、フィルタ、フォントデータ916等は、後述するようにSPサーバー901の電源立上げ時または動作中に、SPサーバー901からの要求に基づき、必要に応じてSPサーバー901にダウンロードされる。

【0562】さらに、ダウンロードホストからSPサーバー901にダウンロードされたフィルタ、フォントデータ916はあらかじめオペレータによって作成されたフィルタ、フォントの情報リストに基づき、新しいフィルタ、フォントがダウンロードされるまでSPサーバー901上のメモリに保持されている。

【0563】また、ダウンロードホストには、あらかじめオペレータによってSPサーバー901の初期設定を行うパラメータファイル917、例えばテキストファイルとして、所定のディレクトリに記憶している。

【0564】さらに、ダウンロードホストにファイルとして記憶されているSPサーバー901のパラメータは、SPサーバー901に接続された端末（図示しない）、あるいは当該スレーブダウンロードホストからパラメータの設定変更を行う場合、SPサーバー901に接続された端末の場合は、当該ダウンロードホストとSPサーバー901の間にEnd-to-Endのセッション通信路を確立し、当該スレーブダウンロードホストの場合は、マスタダウンロードホストとスレーブダウンロードホストとの間にEnd-to-Endのセッション

通信路を確立し、当該マスタダウンロードホストからSPサーバー901あるいはスレーブダウンロードホストにパラメータファイルを転送し、もし、パラメータの変更があれば、当該セッション通信路を通じてマスタダウンロードホストに逆に転送し、所定のディレクトリにパラメータファイルを記憶するプロトコルを有している。

【0565】また、ダウンロードホストにファイルとして記憶されているSPサーバー901のパラメータは、ダウンロードホスト自身から変更することができるように構成されている。

【0566】さらに、特定の1台または複数設定されているダウンロードホストのうち、特定の1台あるいはマスタダウンロードホストが動作していないか、ダウンロード、パラメータファイル転送に失敗した場合、マスタダウンロードホストが1台しかない場合あるいは残りのスレーブホストがない場合、SPサーバー901のエラーログファイルに記録後、あらかじめ設定された最大待ち時間が経過していない場合は、あらかじめ設定されている一定時間毎にダウンロード、パラメータ転送の実行を試み、残りのスレーブホストがある場合は、次の優先度の高いホストコンピュータをマスタダウンロードホストとし、ダウンロード、パラメータ転送の実行を試み、最大待ち時間が経過した場合は、ダウンロード、パラメータ転送失敗となり、エラー終了となるように制御している。

【0567】また、複数台のダウンロードホストが設定され、マスタダウンロードホストのパラメータファイルの変更が行われた場合、マスタダウンロードホストは自動的にスレーブホストとの間にEnd-to-Endのセッション通信路が確立し、変更されたパラメータファイルの転送をすべてのスレーブダウンロードホストに行わせるように制御している。

【0568】さらに、SPサーバー901はネットワーク環境下において、立上げ時ダウンロードホストの自動チェック、ダウンロード、初期設定パラメータ変更時のダウンロードホスト自動変更機構、初期設定パラメータ変更後の複数のダウンロードホストへの自動転送機能を備えている。

【0569】図80は、図78に示したSPサーバー901におけるダウンロードホストチェック処理手順の一例を示すフローチャートである。なお、(1)～(13)は各ステップを示す。

【0570】SPサーバー901の電源がONされると(1)、SPサーバーの不揮発性メモリ901Aに記憶されているダウンロードホストリスト911に基づき、先ず特定の1台であるかまたは複数であるかというダウンロードホスト数が判断される(2)。この判断で台数が複数の場合には、ダウンロードホスト902、903が動作中であるかどうかを判断し(3)、YESならば動作中

で優先度の最も高いホストをマスタダウンロードホストとし(4)、他のダウンロードホストをスレーブダウンロードホストとして(5)、サーバー立上げ時ダウンロードホストチェックを終了する(6)。

【0571】一方、ステップ(3)の判定でNOの場合は、SPサーバーエラーログファイルにその旨を記録し(7)、あらかじめ設定されている最大待ち時間が経過したかどうかを判定し(8)、YESならばサーバー立上げ時ダウンロードホストチェックを終了する(9)。

【0572】一方、ステップ(8)の判定中に、一定時間経過毎に(13)、ステップ(3)に戻り、ダウンロードホストの動作チェックを行う。

【0573】一方、ステップ(2)の判定において、1台の場合は、当該ダウンロードホストが動作中かどうかを判定し(10)、NOならばステップ(7)に戻り、YESならば指定されているダウンロードホストをマスタダウンロードホストとし(11)、サーバー立上げ時ダウンロードホストチェックを終了する(12)。

【0574】図81は、図78に示したSPサーバー901における各種プログラム915、フィルタデータ916のダウンロード処理手順の一例を示すフローチャートである。なお、(1)～(16)は各ステップを示す。

【0575】サーバー電源立上げ時ダウンロードホストチェックが終了すると(1)、SPサーバー901内のダウンロードプログラムまたは動作中その他のSPサーバープログラムによって各種プログラム915、フィルタ、フォントデータ916のダウンロード要求が発生すると(2)、あらかじめSPサーバー901内にあるダウンロードホストの情報リスト、フィルタ、フォントの情報リストがチェックされる(3)。もし、SPサーバー901内にダウンロード要求されたものがあるかどうかを判定し(4)、YESならばステップ(9)に進み、当該記憶されているプログラム、フィルタ、フォントを使用し、ステップ(2)に戻り、次のダウンロード要求が発生するまでダウンロードプログラムはアイドル状態となる。

【0576】一方、ステップ(4)の判定でNOの場合は、現在SPサーバー901内の不揮発性メモリ901A内に残しておくかどうかを情報リストに基づいて判定し、すなわち消去を行うと指定されているかどうかを判定し(5)、NOならばステップ(7)移行に進み、YESならば当該記憶されているプログラム、フィルタ、フォントを消去する(6)。

【0577】次いで、SPサーバー901からマスタダウンロードホストへダウンロード要求がなされ、マスタダウンロードホストからSPサーバーへターゲットのダウンロードが行われる(7)。次いで、ダウンロードが成功したかどうかを判定し(8)、YESならばステップ(9)移行に進み、NOならばSPサーバーエラーログファ

イルに記録し(10)、次のダウンロードホストがあるかどうかを判断し(11)、YESならばそのホストコンピュータをマスタダウンロードホストとし(12)、他のダウンロードホストをスレーブダウンロードホストとし(13)、ステップ(7)に戻る。

【0578】一方、ステップ(11)の判断でNOならば、あらかじめ設定されている最大待ち時間が経過したかどうかを判定し(14)、YESならばダウンロードエラーとして処理を終了する(15)。

【0579】一方、ステップ(14)の判定中に、一定時間経過毎に(16)、ステップ(7)に戻り、プログラム、フィルタ、フォント等のダウンロードを再実行する。

【0580】図82は、図78に示したSPサーバー901における初期設定パラメータファイルの変更処理手順の一例を示すフローチャートである。なお、(1)～(16)は各ステップを示す。また、初期設定パラメータファイル917の変更指示は、SPサーバー901またはスレーブダウンロードホストから初期設定パラメータファイル917を記憶するマスタダウンロードホストに対して行われる。

【0581】SPサーバー901の端末からマスタダウンロードホストに対して初期設定パラメータファイル917の変更指示がなされると(1)、SPサーバー901あるいはスレーブダウンロードホストとの間にセッション通進路が確立される(2)。次いで、通信路が確立されたか(成功)かどうかを判定し(3)、YESならばマスタダウンロードホストからSPサーバー901あるいはスレーブダウンロードホストにパラメータファイルが転送される(4)。次いで、転送が成功したかどうかを判定し(5)、NOならばステップ(10)以降に進み、YESならばSPサーバー901あるいはスレーブダウンロードホストからパラメータの変更が実行される(6)。次いで、変更されたパラメータファイルがSPサーバー901またはスレーブダウンロードホストからマスタダウンロードホストにパラメータファイルの逆転送が行われ(7)、転送が成功したかどうかを判定し(8)、YESならばパラメータ変更後の処理に移行する(9)。

【0582】一方、ステップ(8)、(3)の判定でNOの場合ステップ(10)以降に進み、SPサーバーエラーログファイルにその旨を記録し、次のダウンロードホストがあるかどうかを判定し(11)、YESならば次のダウンロードホストをマスタダウンロードホストとするとともに(12)、他のダウンロードホストをスレーブダウンロードホストとし(13)、ステップ(2)に戻る。

【0583】一方、ステップ(11)の判定でNOの場合は、あらかじめ設定されている最大待ち時間が経過したかどうかを判定し(14)、YESならばパラメータ変更エラーとして処理を終了する(15)。

【0584】一方、ステップ(14)の判定中に、一定時間経過毎に(16)、ステップ(2)に戻り初期設定パラメータ

ファイルの変更処理を再実行する。

【0585】図83は、図78に示したマスタのダウンロードホストからの初期設定パラメータファイルの変更処理手順の一例を示すフローチャートである。なお、

(1)～(3)は各ステップを示す。

【0586】マスタダウンロードホストから初期設定パラメータファイル917の変更指示がなされると(1)、マスタダウンロードホストがマスタダウンロードホスト上の初期設定パラメータファイル917を変更し(2)、変更後の処理に移る(3)。

【0587】図84は、図78に示したマスタのダウンロードホストによる初期設定パラメータファイル変更後処理手順の一例を示すフローチャートである。なお、

(1)～(8)は各ステップを示す。

【0588】初期設定パラメータファイル917が変更されると(1)、マスタダウンロードホストはスレーブダウンロードホストがあるかどうかを判断し(2)、NOならば処理を終了する(8)。

【0589】一方、ステップ(2)の判定においてYESならばマスタダウンロードホストは、スレーブダウンロードホストとの間に通信路を確立し(3)、通信路が確立されたかどうかを判定し(4)、NOならばステップ(7)移行に進み、YESならばマスタダウンロードホストからスレーブダウンロードホストに変更されたパラメータファイルの転送が行われる(5)。次いで、パラメータファイルの転送が成功したかどうかを判定し(6)、YESならばステップ(2)に戻り、NOならばSPサーバーエラーログファイルにその旨を記録し(7)、ステップ(2)に戻り、次のスレーブダウンロードホストを探しに行く。

【0590】なお、上記実施例ではサーバー装置を、リモートプリント、リモートスキャンのサービスを行うSPサーバー901で構成する場合について説明したが、サーバー装置としてはSPサーバー901ではなく、その他の様々な処理を実行するサーバー装置、例えばOCR処理、ファクシミリ処理、ドキュメント処理、データベース処理等をクライアントに提供するサーバー装置であっても、ダウンロードホストを設定したネットワーク環境であれば、同様の方法でダウンロード、パラメータ変更等を行うことができる。

【0591】さらに、サーバー装置に限らず、各々のマシンがネットワーク環境にあり、各々のマシン間のコミュニケーションを行うことが可能であるという条件が成立するならば、上記ダウンロード方式を他の装置にも適用することができる。

【0592】図85は本発明の一実施例を示すサーバー装置の通信制御構成を説明するブロック図である。

【0593】図において、7101A、7101Bはホストコンピュータ(クライアント)で、ネットワーク7105を介してスキャナプリンタサーバー(SPサーバ

ー)7102と通信可能に構成されている。ホストコンピュータ7101A、7101Bは独自のデータ処理速度に応じた通信プロトコル7106A(例えば通信プログラム53)、7106B(例えば通信プログラム63)を備えて、サーバープロトコル7107(例えば(通信プログラム83))と相互に通信処理を後述するフローチャートに従って実行する。SPサーバー7102には各種のプリンタに受信したプリントデータを出力するためのインタフェースポート(RS232C、セントロニクス、SCSI等)を備え、プリンタ7104b、7104c、7104dおよびスキャナ7103bとのI/Oを制御する。7103aはスキャナで、プリンタ7104aと一体としてスキャナ機能処理およびプリント機能処理を実行する。7108～7113は制御ラインで種々の制御信号をプリンタ7104a、スキャナ7103aに出力する。

【0594】この図に示されるように、ネットワーク7105上のホストコンピュータ7101A、7101B(クライアント)がスキャナ機能処理、プリンタ機能処理を実行するために、すなわちスキャナやプリンタの資源を利用するために、SPサーバー7102に問い合わせをする際に情報交換を行う時に、後述する通信処理実行中の応答時間を監視する。その際、応答時間は長いが正常に返信がある通信に対しては、ネットワーク7105上の遅延が生じたが、正常な通信と見なして、自動的にタイマアウトの値を長く設定し直す機能を備えている。

【0595】図86は、図85に示したクライアントとサーバーとの第1の通信プロトコルの一例を示す図であり、特にプリント処理の通信の場合である。

【0596】この図に示すように、経過時間(a)は、ホストコンピュータ7101A側のクライアントプロトコル7106Aは、SPサーバープロトコル7107からの返答がなければ、応答時間監視のためのタイマーT1が作動し、自動的にタイマアウトする時間に対応する。また、経過時間(b)は、SPサーバープロトコル7107がクライアントプロトコル7106Aに対して次の指示あるいはデータが来るまでの応答時間監視時間に対応する。

【0597】このように、本発明に係る第3の発明によれば、サーバー装置から何れかの指示に基づいて通信コマンドを発行後の応答時間が第1の監視時間を越えて、第2の監視時間を経過しても応答が得られない場合、当該ホストとの通信に異常が発生したものと見なして、サーバー装置とネットワークの通信状態を待機状態に自動復帰させるので、当該通信異常となったホストコンピュータの回復を待たずに、プリンタ資源、スキャナ資源による各機能処理実行を要求する他のホストコンピュータとの通信を速やか、かつ正常に開始することが可能となる。

【0598】具体的には、各経過時間(a)、(b)を計時して所定の監視時間を延長しても、応答が無い場合に通信を中断して、待機状態に切り換えることにより、通信状態が互いにデッドロック状態となることを回避している。

【0599】具体的には、SPサーバー7102はネットワーク7105上のホストコンピュータより指示を受け、それに対して通信コマンドを発行する。その際毎(通信コマンド発行毎)に所定の第1のタイマ(例えば5秒タイマ)を起動し、もし、そのタイマが計時終了後も相手からの応答がなかった場合、該通信コマンドを再発行し、さらに再発行が所定回数(例えば6回)または再送コマンドが最初に発行されてからの時間を計時する第2のタイマ(例えば30秒)が計時終了後も相手からの応答がなかった場合に、相手に回復不可能な障害が発生したものと見なし、通信中断コマンド(例えばAbort)を発行して初期状態で待機する。

【0600】実際には、前述したようにSPサーバー7102は、ネットワーク7105上のホストコンピュータ側にクライアントプロトコルが起動されており、SPサーバー7102側にはSPサーバープロトコル7107が起動されている。接続されているスキャナやプリンタの資源のコントロールは、この互いに通信し合うプログラムによっている。

【0601】ところが、SPサーバープロトコル7107が起動されると、図4に示したタイマ110が起動し、所定時間を越えてもSPクライアント(ホストコンピュータ7106A、7106B)から返答がないと、自動的に通信コマンドを再発行して、SPクライアント(ホストコンピュータ7106A、7106B)に通信コマンドに対する返信を要求しても応答がないと、SPサーバープロトコル7107は後述するフローチャートに従って自動的に通信処理プログラムを中断し、初期状態に復帰する。

【0602】なお、本実施例では、上記タイマ処理をリアルタイムOS(VxWorks)のウオッチドックタイマ機能処理により達成しているが、この場合、タイムアウトの状態になると、リアルタイムOS(VxWorks)におけるシステム管理の割込み制御機構により、一旦OS管理下におかれ、すぐにシステム管理を司るSPマネージャ(システム全体制御プログラム93)に通知して、システムの異常処理を後述する図87に示すフローチャートに従って行う。なお、以上の発生する要因として、ネットワーク7105上のホストコンピュータ側が通信異常やシステムダウン、メモリアーオーバーフロー等があげられるが、これらのみには限定されない。

【0603】図87は本発明の一実施例を示すサーバー装置の通信制御方法を説明するフローチャートである。なお、(1)~(11)は各ステップを示し、図85に示したSPサーバー7102のタイマ監視通信制御手順(サー

バープロトコル7107)に対応する。

【0604】ネットワーク7105を介してクライアントからの初期通信要求を待機し(1)、初期通信要求を受信すると、受信した旨をクライアント側に返答し(2)、図4に示したタイマ110を起動して(3)、クライアントからの通信待ちとなる(4)。何らかの通信要求を受信すると、正常通信作業を開始(5)、終了コマンドを受信したかどうかを判定し(6)、NOならば(2)に戻り、YESならばSPサーバー7102を初期状態に設定する。

【0605】一方、ステップ(4)の判定で何ら応答がない場合は、ステップ(3)で起動したタイマがタイムアウト(本実施例では5秒)となったかどうかを判定し(7)、NOならばステップ(4)に戻り、YESならばあらかじめ設定された回数(n)目の再送かどうかを判定し(8)、YESならばOS(本実施例ではVxWorks)の割込み機能を開始し(9)、SPマネージャにその旨を通知して(10)、SPサーバー7102を初期状態に設定する。

【0606】一方、ステップ(8)の判定で、何らかの応答要求があった場合は、再送要求をクライアントに送出し(12)、ステップ(2)に戻る。

【0607】なお、上記実施例では、現在通信中のSPクライアントとSPサーバー7102とが異常の状態となった時、本発明に係る計時手段として機能するタイマ110、内部タイマ等を用いて異常処理を実行し、最終的にはSPサーバー7102が初期状態に戻る場合の処理について説明したが、異常処理を実行すると、別のタスクが起動されて、そのタスクがある一定時間毎にホストコンピュータに通信を行い、ホストコンピュータが復帰することを確認して、通信処理を再開するように制御しても良い。

【0608】また、上記実施例では、通信コマンドを再発行してからの監視時間を、タイマ110がタイムアップする毎に発行される通信コマンドの再送回数(例えば6回)で計時する場合について説明したが、最初の通信コマンド発行後、タイマ110がタイムアップした時点で、第2のタイマ(図示しない)が第2の監視時間(例えば30秒)を計時して、この第2の監視時間の経過時点で、ホストコンピュータからの応答が得られない場合には、通信中断コマンド(Abort)を発行して、通信待機状態となるように制御しても良い。

【0609】図88は、図85に示したクライアントとサーバーとの第2の通信プロトコルの一例を示す図であり、(a)は正常な通信状態に対応し、(b)は応答時間が長い通信状態に対応し、特にプリント処理の通信の場合である。

【0610】(a)に示すように正常な通信時には、経過時間(イ)で、ホストコンピュータ7101A側のクライアントプロトコル7106Aは、SPサーバープロ

トコル 7107 からの返答がなければ、応答時間監視のためのタイマ T1 が作動し、自動的にタイマアウトする。また、経過時間 (ロ) で、SP サーバプロトコル 7107 がクライアントプロトコル 7106A に対して次の指示あるいはデータが来るまでの応答時間監視のタイマ T2 が作動する。正常な通信処理では、ポイント (イ)、(ロ) で計時する監視時間はあらかじめ決められた値が設定されている。もし、それを越える場合は、再送要求を発行して相手からの応答を待機する。それでも応答がない場合には、異常状態とみなして異常処理を行い、初期状態 (通信待ち状態) に移行する。

【0611】一方、(b) に示すように応答時間が長い通信時、例えば遅いネットワークが仲介していたり、トラフィックが多いようなネットワークで通信を行う場合には、経過時間 (イ)、(ロ) は前述と同様のタイマ設定値でタイマが作動する。また、経過時間は、クライアントプロトコル 7106B が SP サーバプロトコル 7107 からの返答が遅い時に、応答要求を発行して、当該返答が帰ってきた時間に応じて、再設定されたタイマのタイマアウト状態に対応する。同様に、経過時間

(二) は、クライアントプロトコル 7106B からの要求に応じて応答した SP サーバプロトコル 7107 の次のコマンドに対してのタイマアウト値が再設定された時間に対応する。

【0612】このように、本発明に係る第 4 の発明によれば、応答時間の長いホストコンピュータ 7106B との通信処理においても、各ホストコンピュータまたはサーバ装置の何れか一方からネットワーク上での応答時間を計時するためのコマンドを発行してからの応答時間をサーバ装置または各ホストコンピュータが計時し、該計時結果に基づいて各ホストコンピュータとサーバ装置との間の正常応答時間を計時するタイマの監視時間を延長するので、ネットワーク上の各ホストコンピュータ上の各ホストとの通信中における応答信号の送出タイミングが長くなる事態が発生しても、その後の通信処理を正常に続行することが可能となる。

【0613】これにより、相手からの応答時間が所定時間よりも経過しても、直ちに通信異常とすることなく、監視時間を所定時間延長して調整することにより、応答時間が長くなる通信処理に対して、通信異常とすることなく、正常な通信処理が可能となる。

【0614】具体的には、SP サーバ 7102 はネットワーク 7105 上のホストコンピュータより指示を受け、それに対して通信コマンドを発行する。その際毎に所定の第 1 のタイマ (例えば 5 秒タイマ) を起動し、もし、そのタイマが計時終了後も相手からの返答がなかった場合、該通信コマンドを再発行し、さらに再発行が所定回数 (例えば 6 回) または再送コマンドが最初に発行されてからの時間を計時する第 2 のタイマ (例えば 30 秒) が計時終了後も相手からの返答がなかった場合に、

相手に回復不可能な障害が発生したものと見なし、通信中断コマンド (例えば Abort) を発行して初期状態で待機する。

【0615】しかしながら、ホストコンピュータと SP サーバ 7102 が異なるネットワークに接続されている場合は、上記タイマのタイムアウト時間 (監視時間) を長く設定する。

【0616】実際には、前述したように SP サーバ 7102 は、ネットワーク 7105 上のホストコンピュータ側にクライアントプロトコルが起動されており、SP サーバ 7102 側には SP サーバプロトコル 7107 が起動されている。接続されているスキャナやプリンタの資源のコントロールは、この互いに通信し合うプログラムによっている。

【0617】本実施例では、SP サーバ 7102 は、SP クライアント (ホストコンピュータ 7106A、7106B) からの資源利用要求 (プリント要求 / スキャン要求) を常に返答を返すが、ネットワークの遅延が大きい場合、SP クライアント側は応答要求を発行して、ネットワークの遅延時間を調べに来る。SP サーバ 7102 側はレスポンス要求を受け取ると、直ちに返答を返す。SP クライアント側は応答要求を発行して、ネットワークの遅延時間を調べに来る。SP サーバ 7102 側はレスポンス応答の時間により、タイマアウトの設定値を変更して、通常の通信を引き続き行う。SP サーバ 7102 は、レスポンス要求を受け付けると、次の通信が来るまでの時間を計時し、SP サーバ 7102 側のタイマアウトの設定値の変更を行う。

【0618】以後、通常動作と同様に、SP サーバ 7102 のプロトコルが起動されると、図 4 に示したタイマ 110 を起動して、所定時間を越えて、SP クライアント側のプロトコルに基づく返答がないと、自動的に再送要求を発行し、それでも返答がないと、自動的に通信の中断を行い、初期状態に戻る。

【0619】なお、本実施例では、上記タイマ処理をリアルタイム OS (VxWorks) のウオッチドックタイマ機能処理により達成しているが、この場合、タイマアウトの状態になると、リアルタイム OS (VxWorks) におけるシステム管理の割り込み制御機構により、一旦 OS 管理下におかれ、すぐにシステム管理を司る SP マネージャに通知して、システムの異常処理を後述する図 89、図 90 に示すフローチャートに従って行う。なお、以上の発生する要因として、ネットワーク 7105 上のホストコンピュータ側が通信異常やシステムダウン、メモリーオーバーフロー等があげられるが、これらの中には限定されない。

【0620】図 89 は本発明の一実施例を示すサーバ装置の通信制御方法を説明するフローチャートである。なお、(1) ~ (14) は各ステップを示し、図 85 に示した SP サーバ 7102 のタイマ監視通信制御手順 (サー

パープロトコル 7107) に対応する。

【0621】ネットワーク 7105 を介してクライアントからの初期通信要求を待機し(1)、初期通信要求を受信すると、受信した旨をクライアント側に返答し(2)、図 4 に示したタイマ 110 を起動して(3)、クライアントからの通信待ちとなる(4)。何らかの通信要求を受信すると、正常通信作業を開始(5)、終了コマンドを受信したかどうかを判定し(6)、NO ならば(2)に戻り、YES ならば SP サーバー 7102 を初期状態に設定する。

【0622】一方、ステップ(4)の判定で何ら応答がない場合は、ステップ(3)で起動したタイマがタイムアウトとなったかどうかを判定し(7)、NO ならばステップ(4)に戻り、YES ならばレスポンス要求有無を判定し(8)、要求が無い場合、あらかじめ設定された回数(n)目の再送かどうかを判定し(11)、YES ならば OS (本実施例では VxWorks) の割込み機能を開始し(13)、SP マネージャにその旨を通知して(14)、SP サーバー 7102 を初期状態に設定する。

【0623】一方、ステップ(8)の判定で、何らかの応答要求があった場合は、タイマ 110 の値を所定時間長く再設定し(9)、応答作業を継続し(10)、ステップ(2)に戻る。また、ステップ(11)の判定で NO の場合は、再送要求をクライアントに送出し(12)、ステップ(2)に戻る。

【0624】図 90 は本発明の一実施例を示すサーバー装置の通信制御方法を説明するフローチャートである。なお、(1)～(16)は各ステップを示し、図 85 に示したクライアントのタイマ監視通信制御手順(クライアントプロトコル 7106A, 7106B)に対応する。

【0625】ネットワーク 7105 を介して SP サーバー 7101 からの初期通信要求を待機し(1)、初期通信要求を受信すると、受信した旨を SP サーバー 7101 側に送信し(2)、ホストコンピュータ 7106A, 7106B の内部タイマを起動し(3)、SP サーバー 7101 からの通信待ちとなる(4)。ここで、何らかの通信を受信すると、受信したコマンドが終了コマンドかどうかを判定し(5)、NO ならば次に通信作業を開始(6)、ステップ(2)に戻り、YES ならば SP クライアントを初期状態とする。

【0626】一方、ステップ(4)の判定で NO の場合は、ステップ(3)で起動したタイマがタイムアウトとなったかどうかを判定し(7)、NO ならばステップ(4)に戻り、YES ならば応答要求が SP サーバー側からなされたかどうかを判定し(8)、YES ならば再送要求がなされたかどうかを判定し(14)、NO ならば SP クライアントのプロトコル 7106A, 7106B に通知し(16)、SP クライアントを初期状態とする。

【0627】一方、ステップ(14)の判定で YES の場合は、再送作業を開始し(15)、ステップ(2)に戻る。ま

た、ステップ(8)の判定で NO の場合は、応答要求を SP サーバー 7102 に対して発行し(9)、応答時間の測定を開始する(10)。次いで、SP サーバー 7102 から応答待ち状態から応答を受信したかどうかを判定し(11)、YES ならばタイマを所定時間長く再設定し(12)、ステップ(2)に戻る。

【0628】一方、ステップ(11)の判定で NO の場合は、応答時間を計時する内部タイマがタイムアウトしたかどうかを判断し(13)、NO ならばステップ(11)に戻り、YES ならば SP クライアントを初期状態に設定する。

【0629】なお、上記実施例では、現在通信中の SP クライアントと SP サーバー 7102 とが異常の状態となった時、計時手段(タイマ 110, 内部タイマ等)を用いて異常処理を実行し、最終的には SP サーバー 7102 が初期状態に戻る場合の処理について説明したが、異常処理を実行すると、別のタスクが起動されて、そのタスクがある一定時間毎にホストコンピュータに通信を行い、ホストコンピュータが復帰することを確認して、通信処理を再開するように制御しても良い。

【0630】図 91 は本発明のカラーレスキャナプリンタサーバーシステムの一実施例の構成を説明する概略ブロック図である。

【0631】図において、9001 はホストマシンで、それぞれの OS に依存したシステムプログラムに基づいてデータ処理を行い、イーサネット(商品名)等のネットワーク 9005 を介してレスキャナプリンタサーバー 9002 と通信可能に構成されている。

【0632】9003, 9004 はレスキャナプリンタとしての画像入出力装置で、画像入出力装置 9003 のプリンタエンジンはレーザープリンタエンジンで構成され、画像入出力装置 9004 はインクジェットプリンタエンジンで構成されている。

【0633】9012 は変換プログラム部で、アプリケーションソフト 9013 から送られたデータおよびデータ構造名に基づく変換プログラムを蓄えている。9011 は通信ソフトで、変換プログラム部 9012 から読み出されたいずれか 1 つの変換プログラムをネットワーク 9005 を介してレスキャナプリンタサーバー 9002 に通信する。

【0634】9021 は通信ソフトで、ネットワーク 9005 上のホストマシン 9001 および図示しないホストマシンとの通信処理を行う。9022 は全体制御ソフトで、レスキャナプリンタサーバー 9002 を総括的に制御する。

【0635】9023 はプリント処理ソフトで、変換ソフト 9023-1 およびプリント処理ソフト 9023-2 とから構成され、変換ソフト 9023-1 は通信された変換プログラムをレスキャナプリンタサーバー 9002 で実行できる形式に変換する。プリント処理ソフト 90

23-2は、変換ソフト9023-1によりプリントデータに変換された内部データを画像入出力装置9003、9004に対応した画像データに変換する。

【0636】9024は画像入力処理ソフトで、ホストマシン9001からの画像入力命令に従って画像入出力装置9003、9004のスキナ部に画像入力を開始するための命令を画像入出力装置制御ソフト9025、9026に出力し、画像入出力装置9003、9004から出力される画像データを全体制御ソフト9022、通信ソフト9021の制御の基にネットワーク9005を介してホストマシン9001に出力する。

【0637】なお、通信ソフト9011は、アプリケーションソフト9013のデータ構造からスキナプリンタサーバ9002が受け付けるデータ構造に変換するための変換アルゴリズムが入った特定の変換プログラムを変換プログラム部9012のなかの1つより選択して、スキナプリンタサーバ9002側の通信ソフト9021に送信するが、ここでアプリケーションソフト9013がウインドウズ（商品名）であれば、GDIからCaPSLデータに変換し、マッキントッシュ（Macintosh（商品名））であれば、クイックドロ（QuickDraw）からCaPSLデータに変換するための特定の変換プログラムを選択する。ちなみに、本実施例で記述するCaPSLデータとは、スキナプリンタサーバ9002が受け付けるデータ構造を指している。

【0638】以下、ホストマシン9001からのスキナプリンタ要求時の第1のデータ処理動作について説明する。

【0639】全体制御ソフト9022は、ネットワーク9005を介して送られてきたデータを解析し、そのときのスキナプリンタサーバ9002の状態と送られてきたデータに従い処理を行う。

【0640】例えば送られてきたデータがプリント命令である場合は、プリントデータをプリント処理ソフト9023に送る。プリント処理ソフト9023は、プリントデータおよび変換プログラムを受け取る。そして、先ず、スキナプリンタサーバ9002側で受けるデータ形式に変換ソフト9023-1で変換する。変換ソフト9023-1は変換プログラムをスキナプリンタサーバ9002で実行できる形式にしたものである。従って、変換ソフト9023-1ではGDIからCaPSL（商品名）またはQuickDraw（商品名）からCaPSLへの変換等が行われる。そして、プリント処理ソフト9023の1部分として構成されるプリント処理ソフト9023-2は変換されたプリントデータを受け取り、画像入出力装置9003、9004が受け入れ可能なデータに変換する。例えばPostScript（商品名）やCaPSLといったページ記述言語のデータから画像データに変換する。

【0641】全体制御ソフト9022は、プリント処理ソフト9023（9023-2）が変換した画像データを画像入出力装置制御ソフト9025、9026に送り、画像入出力装置制御ソフト9025、9026は画像データを画像入出力装置9003、9004に送りプリントされる。

【0642】また、送られてきたデータが画像入力命令であった場合には、画像入力命令を画像入力処理ソフト9024に送り、画像入力処理ソフトは9024命令に従って画像入出力制御ソフト命令を送り、画像入出力装置を起動して画像データを入力し、画像データを画像入力処理ソフト9024へ送り、画像データは全体制御ソフト9022を介して通信ソフト9021へ送られ、スキナプリンタサーバ9002よりホストマシン9001の通信ソフト9011へ画像データが通信され、さらに、アプリケーションソフト9013に送られる。

【0643】一方、スキナプリンタサーバ9002の状態を問い合わせる命令が送られた場合は、全体制御ソフト9022がスキナプリンタサーバ9002を通信ソフト9021を通じてホストマシン9001に通信する。同様に、画像入出力装置9003、9004やスキナプリンタサーバ9002内でエラーが発生した場合は、全体制御ソフト9022がエラー状況を管理し、通信ソフト9021を通じてホストマシン9001に連絡する。

【0644】各構成要素の主な機能をまとめてみると、アプリケーションソフト9013は、通信ソフト9011にプリントや画像入力や状況問い合わせ等の命令を送り、通信ソフト9011はスキナプリンタサーバ9002の間に通信を行う。全体制御ソフト9022はスキナプリンタサーバ9002全体の制御を行う。プリント処理ソフト9023は、スキナプリンタサーバ9002で扱うPDL（PostScript（商品名）やCaPSL（商品名））形式への変換を行いプリントデータを作成するとともに、プリントデータから各画像入出力装置9003、9004が受け入れられる画像データへの変換を行う。画像入力処理ソフト9024は、画像入出力装置9003、9004を使って画像データを入力してホストマシン9001へ送る処理を行う。

【0645】画像入出力装置制御ソフト9025、9026は、画像入出力装置9003、9004を制御し、画像入出力装置9003、9004はプリントやスキナ等の画像入出力処理を行う。

【0646】本実施例では、これらのソフトウェア構成により、ホストマシン9001のアプリケーションプログラムからスキナプリンタサーバ9002の指定したプリンタでプリントしたり、スキナから画像入力したり、スキナプリンタサーバ9002の状態を確認することができる。

【0647】以下、図91を参照しながらスキナプリ

ンタ要求時の第2のデータ処理動作について説明する。

【0648】データ構造名に基づく変換プログラムをホストマシン9001より、スキナプリンタサーバ9002に対してシステム立ち上げ時にあらかじめ通信ソフト9011より通信ソフト9021に送っても良い。

【0649】また、スキナプリンタサーバ9002に対してシステム起動後に行うプリントの時に必要な変換プログラムが初めて使われる時に1回だけ、通信ソフト9011より通信ソフト9021に送り、プリント命令毎に必要な変換プログラムが送られているか、双方で確認するか、自システムの記録を参照して確認し、必要な変換プログラムがまだ送られていない場合のみ変換プログラムを送るような構成としてもよい。

【0650】なお、場合によっては、クライアントであるホスト側でデータ構造名に基づく変換プログラムで変換をホスト側で行い、サーバ側の変換ソフト9023-3を省略する形態として構成される場合もある。この場合には、スキナプリンタサーバ9002側の能力がクライアント側と比較して低い時に有効となり、通信ソフト9011および通信ソフト9021間で決定選択できるように構成すればよい。

【0651】以下、ホストマシン9001からのスキナプリンタ要求時の第3のデータ処理動作について説明する。

【0652】クライアントが持つ変換プログラム部9012をあらかじめスキナプリンタサーバ9002の中に持って常駐させ、変換ソフト9023-1内に持つ構成としてもよい。

【0653】この場合、スキナプリンタサーバ9002は、予め予想される代表的な数種類の変換プログラムを持ち、もし、スキナプリンタサーバ9002に変換プログラムを持たない時に限り、クライアント側（ホスト側）より変換プログラムの供給を受ける。

【0654】また、スキナプリンタサーバ9002側では、ホスト側より送られるデータのデータ構造種類を解析する機能を持つか、またはホスト側よりデータおよびデータの構造種類を受けて、対応する変換プログラムを起動させる構成としてもよい。

【0655】以下、本実施例と第1～第9の発明の各手段との対応及びその作用について図91等を参照しながら説明する。

【0656】第1の発明は、スキナ機能処理およびプリンタ機能処理を実行する入出力機器（画像入出力装置9003、9004）が所定のインタフェースを介して接続され、かつ所定のネットワーク9005を介してオペレーティングシステムが複数のホストマシン9001と通信可能に接続されたスキナプリンタサーバ9002とから構成されるスキナプリンタサーバシステムにおいて、各ホストのオペレーティングシステム（OS）に対応したデータ構造を有する印刷データを前記ス

キナプリンタサーバ装置内のページ記述言語（本実施例ではCaSPLEとするがポストスクリプト（商品名）であってもよい）に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段（通信ソフト9011による）を各ホストマシンに設け、各ホストマシンの転送手段から転送される各変換プログラムファイルに基づいてスキナプリンタサーバ装置本体内のページ記述言語に対応する印刷データに変換する変換手段（変換ソフトを含むプリント処理ソフト9023）をスキナプリンタサーバ9002に設け、各ホストの通信ソフト9011から各ホストのオペレーティングシステムに対応したデータ構造（上記クイックドロ、GDI等）を有する印刷データを前記スキナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル（変換プログラム部9012内に記憶されている）がスキナプリンタサーバ装置本体に転送されたら、該転送された各変換プログラムファイルに基づいてプリント処理ソフト9023が各ホストから転送される印刷データをスキナプリンタサーバ装置内のページ記述言語に対応する印刷データに変換して、各ホスト側によるスキナプリンタサーバ装置内のページ記述言語に対応する印刷データへの変換処理をスキナプリンタサーバ装置側で実行して、各ホストによる印刷データ転送処理負担を軽減する。

【0657】第2の発明は、転送手段（通信ソフト9011）が起動しているアプリケーションのデータ構造に基づいて記憶手段から選択した変換プログラムファイルを転送して、各アプリケーションに対応した最適な変換プログラムファイルをスキナプリンタサーバ装置に転送する。

【0658】第3の発明は、転送手段（通信ソフト9011）は、印刷要求毎に印刷データとともに変換プログラムファイルを転送して、印刷要求時に対応する変換プログラムファイルを印刷データと対して確実に転送する。

【0659】第4の発明は、転送手段（通信ソフト9011）は、ネットワーク初期設定時に、記憶手段に記憶されたいずれかの変換プログラムファイルを転送して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0660】第5の発明は、転送手段（通信ソフト9011）は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を確認して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0661】第6の発明は、転送手段（通信ソフト9011）は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無をスキナプリンタサーバ装置に問い合わせ確認し

て、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0662】第7の発明は、転送手段（通信ソフト9011）は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を転送履歴情報に基づいて確認して、各ホストからの同一の変換プログラムファイル転送負担を軽減する。

【0663】第8の発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器（画像入出力装置9003、9004）が所定のインタフェースを介して接続され、かつ所定のネットワーク9005を介してオペレーティングシステムが複数のホストマシン9001と通信可能に接続されたスキャナプリンタサーバー装置9002とから構成されるスキャナプリンタサーバーシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルを転送する転送手段（通信ソフト9011）を各ホストに設け、各ホストの転送手段から転送された変換プログラムファイルを記憶する記憶手段（ハードディスク（HD）9006）と、前記転送手段から転送される印刷データを解析して前記記憶手段に対応する変換プログラムファイルが登録されているかどうかを判定する判定手段（全体制御ソフト9022による）と、この判定手段による判定結果に基づいて解析された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求する変換ファイル転送要求手段（通信ソフト9021による）と、各ホストの転送手段から転送される印刷データを記憶手段に記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換する変換手段（変換ソフトを含むプリント処理ソフト9023）とをスキャナプリンタサーバー9002に設け、各ホストの通信ソフト9011から転送された変換プログラムファイルをハードディスク（HD）9006に記憶しておき、全体制御ソフト9022が各ホストの通信ソフト9011から転送される印刷データを解析してハードディスク（HD）9006に対応する変換プログラムファイルが登録されているかどうかを判定し、該判定結果に基づいて通信ソフト9021が解析された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求し、各ホストの通信ソフト9011から転送される印刷データをハードディスク（HD）9006に記憶し、各ホストから転送される印刷データをプリント処理ソフト9023が記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサ

ーバー装置内のページ記述言語に対応する印刷データに変換して、各ホストからの印刷要求時におけるスキャナプリンタサーバー装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮する。

【0664】第9の発明は、変換手段（プリント処理ソフト9023）は、解析された印刷データの形式に従って記憶手段（ハードディスク（HD）9006）に登録されたいずれかの変換プログラムファイルを選択してスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換して、各ホストからの印刷要求時におけるスキャナプリンタサーバー装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮する。

【0665】以下、図92、図93に示すフローチャートを参照しながら本発明に係るスキャナサーバープリンタシステムのデータ処理方法について説明する。

【0666】図92は本発明に係るスキャナサーバープリンタシステムの第1のデータ処理方法の一実施例を示すフローチャートである。なお、（1）～（4）は各ステップを示す。

【0667】まず、通信ソフト9011は印刷要求が図示しないキーボード等の操作によりなされるのを待機し（1）、印刷要求がなされたら、印刷データの構造を解析して（2）、変換プログラム部9012から最適な変換プログラムファイルを選択し（3）、選択された変換プログラムファイルおよび印刷データの転送を開始し、該転送終了となったかどうかを判定して（4）、転送終了となったら処理を終了する。

【0668】図93は本発明に係るスキャナサーバープリンタシステムの第2のデータ処理方法の一実施例を示すフローチャートである。なお、（1）～（6）は各ステップを示す。

【0669】ネットワーク9005を介していずれかのホストマシン9001より印刷データを受信したら

（1）、当該印刷データに対する変換プログラムファイルの転送を待機し（2）、当該印刷データを転送したホストマシンより変換プログラムファイルを受信したら、プリント処理ソフト9023が、受信した変換プログラムに基づいてクイックドロウ等の印刷データが終了するまで、例えばCaPSLに変換し（3）、（4）、CaPSLへの変換が終了したら、画像入出力装置制御ソフト9025がCaPSLデータを所定のイメージデータに変換し（5）、画像入出力装置9003または画像入出力装置9004のいずれかで印刷し（6）、印刷終了を各ホストに通知して、処理を終了する。

【0670】なお、各ホストマシン2は、ネットワーク初期設定時に、スキャナプリンタサーバー9002と所定のデータ転送に伴う処理時間を経時して、あるいはスキャナプリンタサーバーに登録された処理能力データを読み出して、印刷データの転送時に変換プログラムファイルの転送を行うか、ホスト側で変換プログラムに基づ

いてスキャナプリンタサーバー 9002 上の CAPSL データに変換するかを選択し、スキャナプリンタサーバー 9002 の処理能力に応じた印刷データを処理を行えるように構成してもよい。

【0671】以下、本実施例と第 10 の発明の各手段との対応及びその作用について図 92 を参照しながら説明する。

【0672】第 10 の発明は、スキャナ機能処理およびプリンタ機能処理を実行する入出力機器（画像入出力装置 9003、9004）が所定のインタフェースを介して接続され、かつ所定のネットワーク 9005 を介してオペレーティングシステムが複数のホストマシン 9001 と通信可能に接続されたスキャナプリンタサーバー 9002 とから構成されるスキャナプリンタサーバーシステムにおいて、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルをいずれかのホストからスキャナプリンタサーバー装置に転送する転送工程（図 92 のステップ（3）、（4））を実行して、スキャナプリンタサーバー 9002 が各ホストから転送される各変換プログラムファイルに基づいて各ホストから転送された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換して、従来各ホストで実行されていたスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換処理をスキャナプリンタサーバー装置に移行させて、各ホストによる印刷データ転送処理負担を格段に軽減する。

【0673】

【発明の効果】以上説明したように、本発明に係る第 1 の発明によれば、各ホストの転送手段から各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルがスキャナプリンタサーバー装置に転送されたら、該転送された各変換プログラムファイルに基づいて各ホストから転送される印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換して、各ホスト側によるスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データへの変換処理をスキャナプリンタサーバー装置側で実行するので、各ホストによる印刷データ転送処理負担を軽減することができる。

【0674】第 2 の発明によれば、転送手段が起動しているアプリケーションのデータ構造に基づいて記憶手段から選択した変換プログラムファイルを転送するので、各アプリケーションに対応した最適な変換プログラムファイルをスキャナプリンタサーバー装置に転送することができる。

【0675】第 3 の発明によれば、転送手段は、印刷要求毎に印刷データとともに変換プログラムファイルを転送するので、印刷要求時に対応する変換プログラムファイルを印刷データと対として確実に転送することができる。

【0676】第 4 の発明によれば、転送手段は、ネットワーク初期設定時に、記憶手段に記憶されたいずれかの変換プログラムファイルを転送するので、各ホストからの同一の変換プログラムファイル転送負担を軽減することができる。

【0677】第 5 の発明によれば、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を確認して、各ホストで同一変換プログラムファイルの既転送状態を管理しながら、各ホストからの同一の変換プログラムファイル転送負担を軽減することができる。

【0678】第 6 の発明によれば、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無をスキャナプリンタサーバー装置に問い合わせ確認して、各ホストとスキャナプリンタサーバー装置とが相互に同一変換プログラムファイルの既転送状態を管理しながら、各ホストからの同一の変換プログラムファイル転送負担を軽減することができる。

【0679】第 7 の発明によれば、転送手段は、要求されている変換プログラムファイル転送前に、同一変換プログラムファイルの既転送状態有無を転送履歴情報に基づいて確認して、各ホストで同一変換プログラムファイルの既転送状態を各転送履歴を管理しながら、各ホストからの同一の変換プログラムファイル転送負担を軽減することができる。

【0680】第 8 の発明によれば、各ホストの転送手段から転送された変換プログラムファイルを記憶手段に記憶しておき、判定手段が各ホストの転送手段から転送される印刷データを解析して前記記憶手段に対応する変換プログラムファイルが登録されているかどうかを判定し、該判定結果に基づいて変換ファイル転送要求手段が解析された印刷データをスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイル転送を対応するいずれかのホストに要求し、各ホストの転送手段から転送される印刷データを記憶手段に記憶し、各ホストから転送される印刷データを変換手段が記憶されたいずれかの変換プログラムファイルに基づいてスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するので、各ホストから転送される変換プログラムファイルをサーバー装置側で管理しながら、各ホストからの印刷要求時におけるスキャナプリンタサーバー装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮することができる。

【0681】第9の発明によれば、変換手段は、解析された印刷データの形式に従って記憶手段に登録されたいずれかの変換プログラムファイルを選択してスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するので、各ホストから転送される印刷データを最適な変換プログラムファイルを選択できるようにサーバー装置側で転送された変換プログラムファイルを識別管理しつつ、各ホストからの印刷要求時におけるスキャナプリンタサーバー装置に対する転送データ量を軽減し、印刷データ転送を時間を短縮することができる。

【0682】第10の発明によれば、各ホストのオペレーティングシステムに対応したデータ構造を有する印刷データを前記スキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換するための変換プログラムファイルをいずれかのホストからスキャナプリンタサーバー装置に転送するので、従来各ホストで実行されていたスキャナプリンタサーバー装置内のページ記述言語に対応する印刷データに変換処理をスキャナプリンタサーバー装置に移行させて、各ホストによる印刷データ転送処理負担を格段に軽減することができる。

【0683】従って、各ホストからスキャナプリンタサーバー装置への印刷データ転送時における各ホストの転送処理負担を格段に軽減して、各ホストのデータ処理能力低下を防止して快速なデータ処理環境を維持できる等の効果を奏する。

【図面の簡単な説明】

【図1】本発明の第1実施例を示す画像処理システムにおけるサーバー装置の概要を説明するシステムブロック図である。

【図2】本発明の第1実施例を示す画像処理システムの一例を示すシステムブロック図である。

【図3】図2に示した画像処理システムにおけるサーバー装置と各ホストとのネットワーク構築状態を示す図である。

【図4】図1に示したメインCPUボード回路の詳細構成を説明する回路ブロック図である。

【図5】図4に示したメモリクリアコントローラの詳細構成を説明するブロック図である。

【図6】図5に示したバンドメモリへの画像情報のバンド展開処理を説明する模式図である。

【図7】図5に示したバンドメモリへの画像情報のバンド展開処理を説明する模式図である。

【図8】図5に示した画像描画処理回路の詳細構成を説明するブロック図である。

【図9】図1に示した第1のインタフェース回路の内部構成を説明するブロック図である。

【図10】図9に示したリアルタイム圧縮伸長部の詳細構成の一例を示すブロック図である。

【図11】図10に示したジグザグスキャン部によるジ

グザグスキャン経路の一例を示す図である。

【図12】図1に示した第1のSPインタフェース回路のパート構成の概略を説明するブロック図である。

【図13】図1に示した第1のSPインタフェース回路の詳細内部構成を説明する回路ブロック図である。

【図14】図1に示した第1のSPインタフェース回路の詳細内部構成を説明する回路ブロック図である。

【図15】図1に示した第1のSPインタフェース回路の詳細内部構成を説明する回路ブロック図である。

【図16】図1に示した第2のSPインタフェース回路の詳細内部構成を説明する回路ブロック図である。

【図17】図2に示したスキャナ/プリンタの画像記録プロセスを説明する模式図である。

【図18】図2に示したスキャナ/プリンタのスキャナ原稿走査状態を示す模式図である。

【図19】図2に示したスキャナ/プリンタのスキャナバンド原稿走査状態を示す模式図である。

【図20】本発明に係るスキャナプリンタサーバーシステムとプリンタとのインタフェース信号の一例を示す図である。

【図21】本発明に係るスキャナプリンタサーバーシステムとプリンタとのインタフェースの一例を示す回路ブロック図である。

【図22】図21の動作を説明するタイミングチャートである。

【図23】図21に示したセントロニクスI/F回路によるホスト側の信号処理手順の一例を示すフローチャートである。

【図24】図21に示したセントロニクスI/F回路によるプリンタ側の信号処理手順の一例を示すフローチャートである。

【図25】本発明に係るスキャナプリンタサーバーシステムとホストコンピュータとのプログラム構成を説明する図である。

【図26】本発明に係るスキャナプリンタサーバーシステムにおける全体制御手順の一例を示すフローチャートである。

【図27】本発明に係るスキャナプリンタサーバーシステムにおけるページ記述言語に準拠するコード体系の要部を示す図である。

【図28】本発明に係るスキャナプリンタサーバーシステムにおけるレイアウトによる中間コードへの置換処理を説明する図である。

【図29】本発明に係るスキャナプリンタサーバーシステムにおけるバンドメモリへの図形情報展開処理状態を示す模式図である。

【図30】本発明に係るスキャナプリンタサーバーシステムにおけるページ記述言語が展開するデータの1画素構造を説明する図である。

【図31】図2に示したスキャナの動作を説明するデー

タ処理経路図である。

【図 3 2】本発明に係るスキャナプリンタサーバシステムとカラーレーザ複写装置とのデータ処理状態を説明するブロック図である。

【図 3 3】本発明に係るスキャナプリンタサーバシステムとカラーレーザ複写装置とのデータ処理状態を説明するブロック図である。

【図 3 4】本発明に係るスキャナプリンタサーバシステムとバブルジェットカラー複写装置とのデータ処理状態を説明するブロック図である。

【図 3 5】本発明に係るスキャナプリンタサーバシステムとのネットワークシステムの構成を説明するブロック図である。

【図 3 6】本発明に係るスキャナプリンタサーバシステムにおける異常発生ジョブ処理手順の一例を示すフローチャートである。

【図 3 7】本発明に係るスキャナプリンタサーバシステムにおけるリトライ処理手順の一例を示すフローチャートである。

【図 3 8】本発明に係るスキャナプリンタサーバシステムにおけるジョブ制御状態を説明するタイミングチャートである。

【図 3 9】本発明に係るスキャナプリンタサーバシステムにおけるシステム全体制御プログラム制御処理状態を示す図である。

【図 4 0】図 3 9 に示したシステム全体制御プログラムのイベント処理手順の一例を示すフローチャートである。

【図 4 1】本発明に係るスキャナプリンタサーバシステムにおけるジョブ管理状態を示す図である。

【図 4 2】本発明に係るスキャナプリンタサーバシステムにおけるジョブ実行処理手順の一例を示すフローチャートである。

【図 4 3】本発明に係るスキャナプリンタサーバシステムにおけるステータスチェック処理手順の一例を示すフローチャートである。

【図 4 4】本発明に係るスキャナプリンタサーバシステムにおける優先順位チェック処理手順の一例を示すフローチャートである。

【図 4 5】本発明に係るスキャナプリンタサーバシステムにおけるインタフェースチェック処理手順の一例を示すフローチャートである。

【図 4 6】本発明に係るスキャナプリンタサーバシステムにおけるジョブ起動処理手順の一例を示すフローチャートである。

【図 4 7】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理状態推移を示すタイミングチャートである。

【図 4 8】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を

示す図である。

【図 4 9】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 0】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 1】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 2】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 3】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 4】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 5】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 6】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 7】本発明に係るスキャナプリンタサーバシステムにおけるジョブ処理に伴うジョブテーブルの内容を示す図である。

【図 5 8】本発明の第 2 の実施例を示すサーバ装置の構成を説明するブロック図である。

【図 5 9】図 5 8 に示した SP サーバ装置により合成される印刷レイアウトを示す模式図である。

【図 6 0】図 2 に示した第 1 のステーションのソフトウェア構成を説明する図である。

【図 6 1】図 6 0 に示した第 2 のステーションの OS と各ドライバとの関係を示す詳細図である。

【図 6 2】図 6 0 に示したプリンタマネージャーとプリンタドライバとの機能処理を説明する図である。

【図 6 3】図 2 に示した第 2 のステーションにおけるネットワークプリンタのプロトコルを説明する図である。

【図 6 4】図 2 に示した第 2 のステーションによる印刷プロセスの一例を示すシーケンス図である。

【図 6 5】図 6 3 に示した SP クラアントのプログラムの構造を示す図である。

【図 6 6】図 2 に示した第 2 のステーションと SP サーバとのネットワーク接続状態を説明する図である。

【図 6 7】図 2 に示した第 2 のステーション上のウィンドウズと各ドライバとの関係を説明する図である。

【図 6 8】図 2 に示した第 2 のステーション上と SP サーバとの各ドライバとの関係を説明する図である。

【図 69】図 2 に示した各ホストのネットワークプリンタ処理プロセスを説明する概念図である。

【図 70】本発明に係るスキャナプリンタサーバシステムの SP サーバと各ホストの SP クライアントとの通信プロセスを説明する概念図である。

【図 71】図 3 に示した SP サーバによるパケット処理手順の一例を示す図である。

【図 72】本発明に係るスキャナプリンタサーバシステムに接続されるプリンタにおけるカラー／白黒印刷制御手順の一例を示すフローチャートである。

【図 73】本発明に係るスキャナプリンタサーバシステムにおける印刷ジョブの流れを説明する図である。

【図 74】図 73 に示した白黒印刷吸収条件を示す図である。

【図 75】図 2 に示した SP サーバ装置と第 3 のステーションとのネットワークシステムの概要を説明するブロック図である。

【図 76】図 75 に示したワークステーションの機能を説明するブロック図である。

【図 77】図 75 に示した SP サーバの機能構成を説明するブロック図である。

【図 78】本発明に係るスキャナプリンタサーバシステムと各ホストとのデータロード処理状態を説明するネットワーク図である。

【図 79】図 78 に示した SP サーバおよびダウンロードホストの記憶情報を説明するデータ構成図である。

【図 80】図 78 に示した SP サーバにおけるダウンロードホストチェック処理手順の一例を示すフローチャートである。

【図 81】図 78 に示した SP サーバにおけるプログラム、フォント、フィルタのダウンロード処理手順の一例を示すフローチャートである。

【図 82】図 78 に示した SP サーバにおける初期設定パラメータファイルの変更処理手順の一例を示すフローチャートである。

【図 83】図 78 に示したマスタのダウンロードホストからの初期設定パラメータファイルの変更処理手順の一例を示すフローチャートである。

【図 84】図 78 に示したマスタのダウンロードホスト

からの初期設定パラメータファイルの変更後処理手順の一例を示すフローチャートである。

【図 85】本発明の一実施例を示すサーバ装置の通信制御構成を説明するブロック図である。

【図 86】図 85 に示したクライアントサーバ装置と第 1 の通信プロトコルの一例を示す図である。

【図 87】本発明の一実施例を示すサーバ装置の通信制御方法を説明するフローチャートである。

【図 88】図 85 に示したクライアントとサーバ装置と第 2 の通信プロトコルの一例を示す図である。

【図 89】本発明の一実施例を示すサーバ装置の第 1 の通信制御方法を説明するフローチャートである。

【図 90】本発明の一実施例を示すサーバ装置の第 2 の通信制御方法を説明するフローチャートである。

【図 91】本発明のカラースキャナプリンタサーバシステムの一実施例の構成を説明する概略ブロック図である。

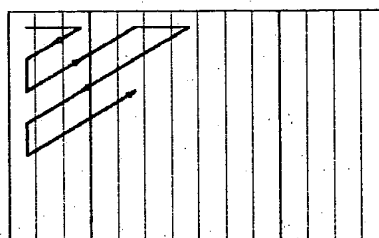
【図 92】本発明に係るスキャナサーバプリンタシステムの第 1 のデータ処理方法の一実施例を示すフローチャートである。

【図 93】本発明に係るスキャナサーバプリンタシステムの第 2 のデータ処理方法の一実施例を示すフローチャートである。

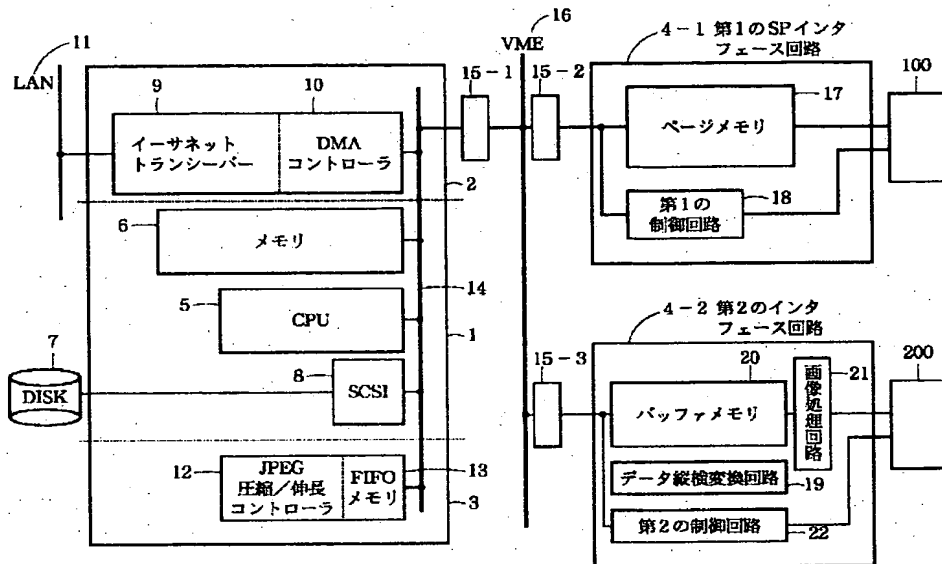
【符号の説明】

- 9001 ホストマシン
- 9002 スキャナプリンタサーバ
- 9003 画像入出力装置
- 9004 画像入出力装置
- 9005 ネットワーク
- 9006 ハードディスク (HD)
- 9011 通信ソフト
- 9012 変換プログラム部
- 9013 アプリケーションソフト
- 9021 通信ソフト
- 9022 全体制御ソフト
- 9023 プリント処理ソフト
- 9024 画像入力処理ソフト
- 9025 画像入出力装置制御ソフト
- 9026 画像入出力装置制御ソフト

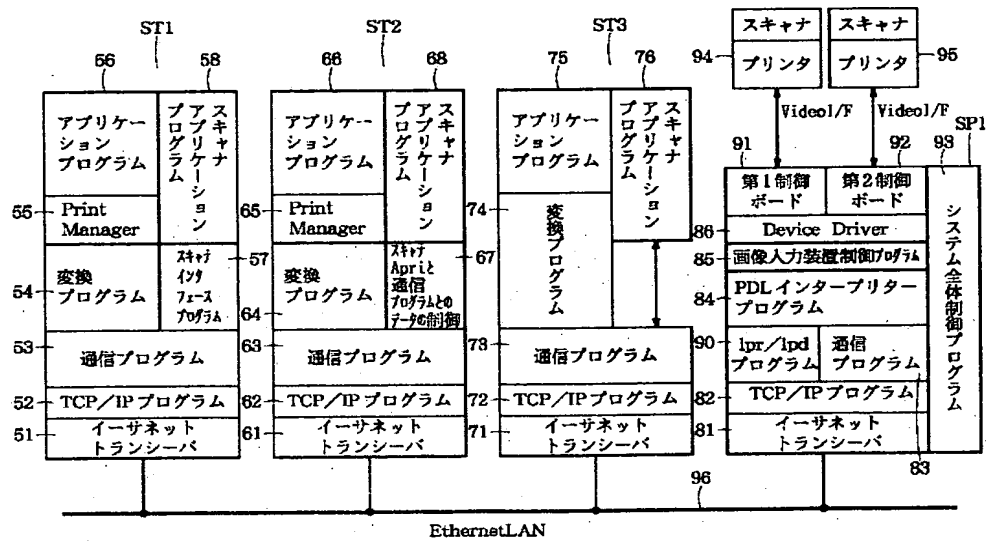
【図 11】



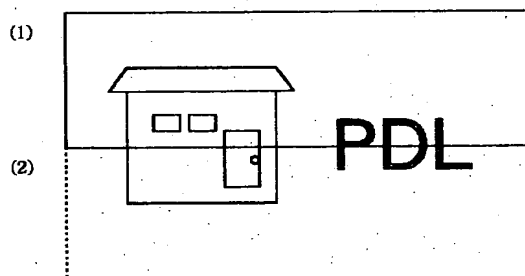
【図 1】



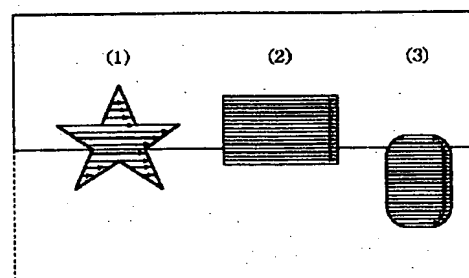
【図 2】



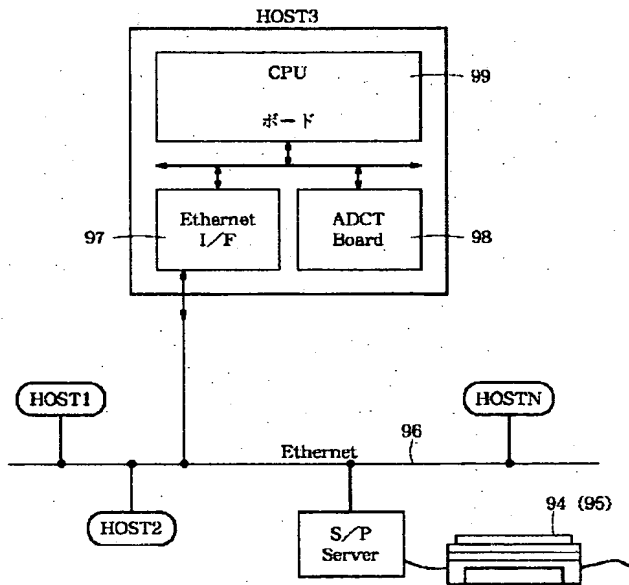
【図 6】



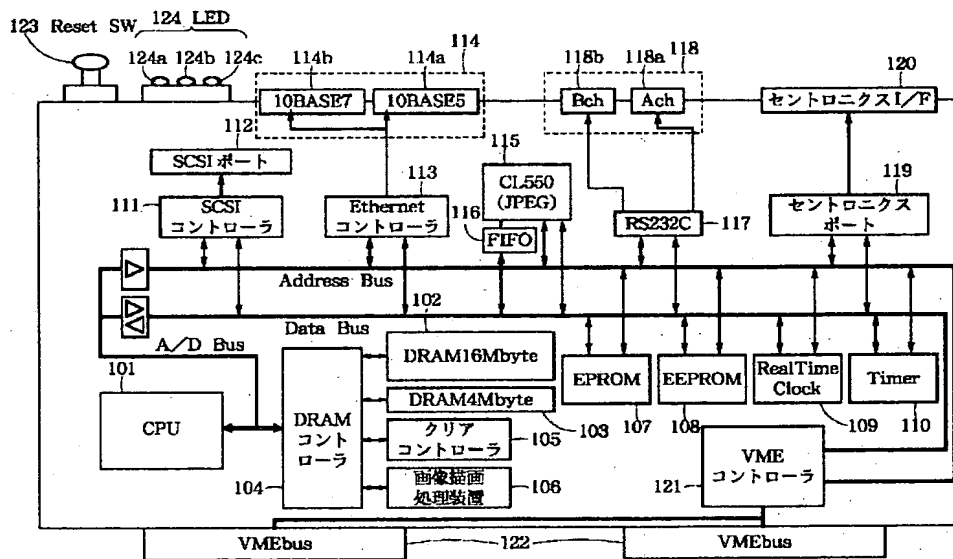
【図 7】



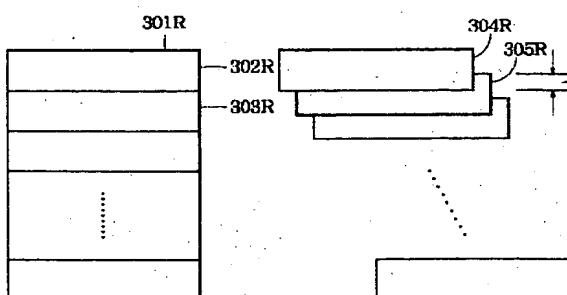
【図 3】



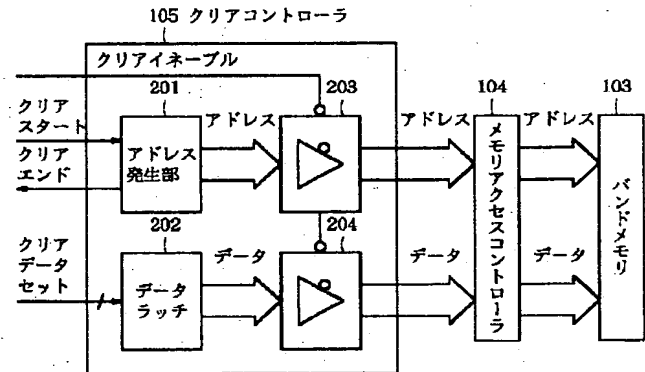
【図 4】



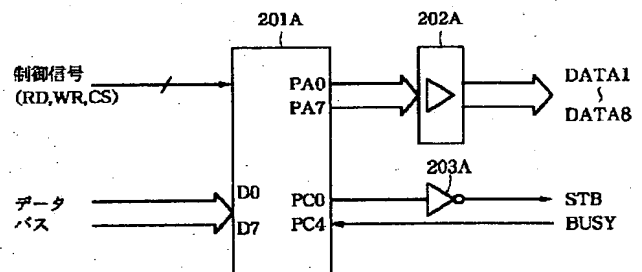
【図 19】



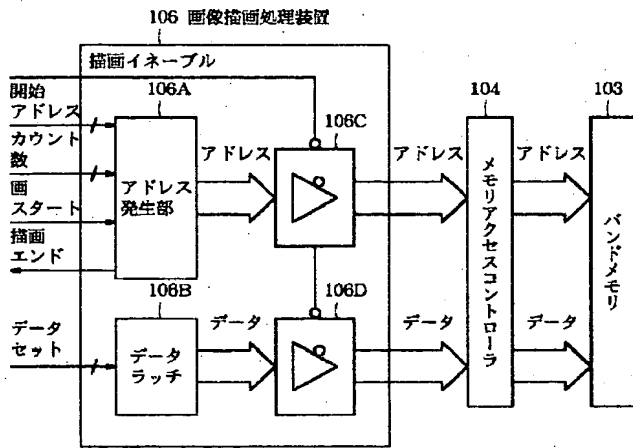
【図 5】



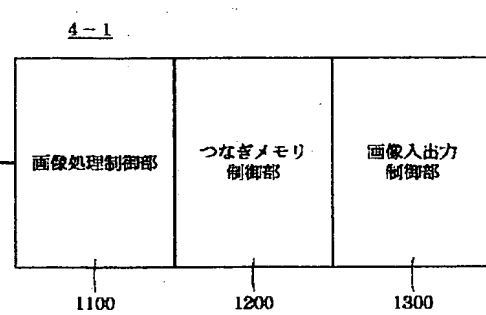
【図 21】



【図 8】

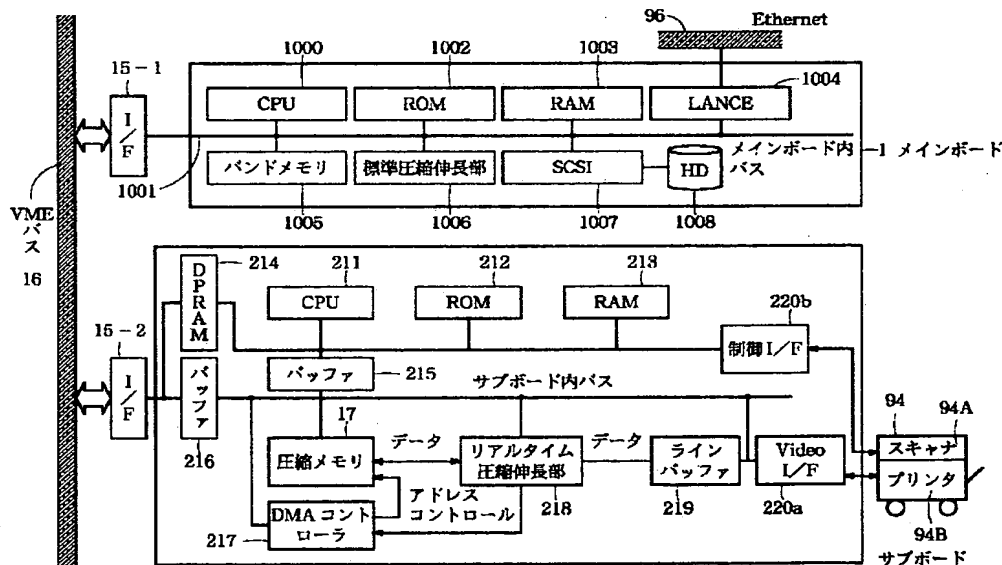


【図 12】



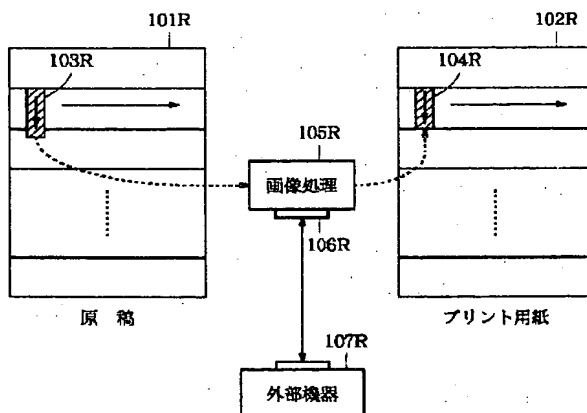
【図 29】

【図 9】



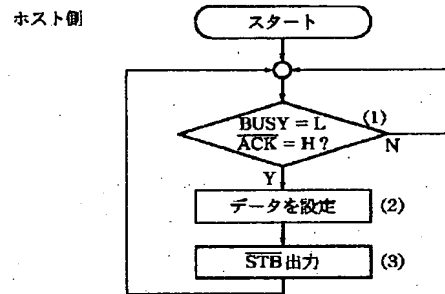
【図 17】

【図 20】

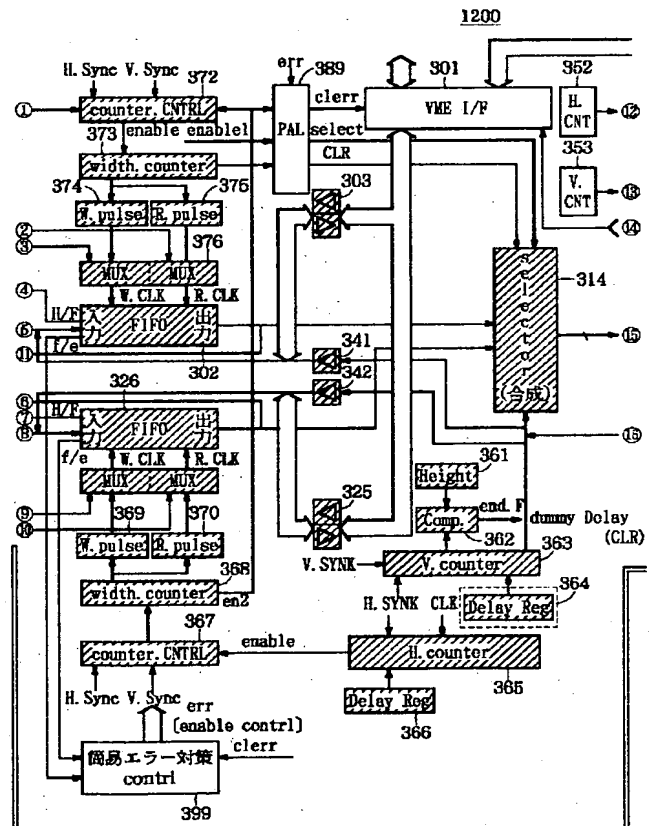


信号名	入出力区分 (プリンタ側)	備考
DATA STB	入力	
DATA1	入力	
DATA8		
ACK		
BUSY	出力	
PE	出力	Paper End (紙切れ)
SERECT	出力	プリンタのセレクト
SG		Signal Ground
FG		Frame Ground
+5V		
INPUT PRIME	入力	初期化 (プリンタを初期化する)
FAULT	入力	エラー
SG		グラウンド

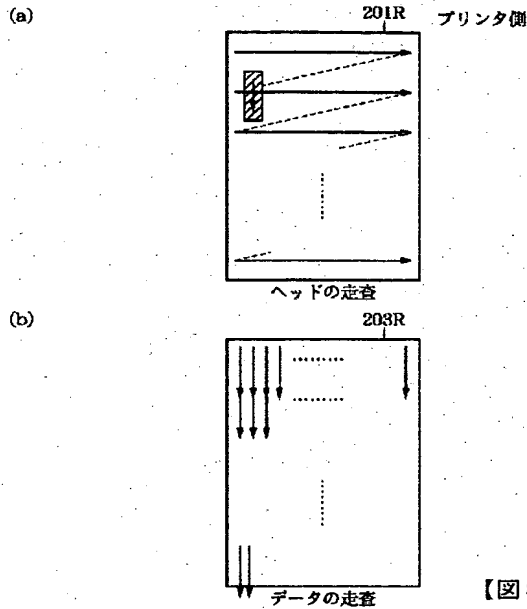
【圖 23】



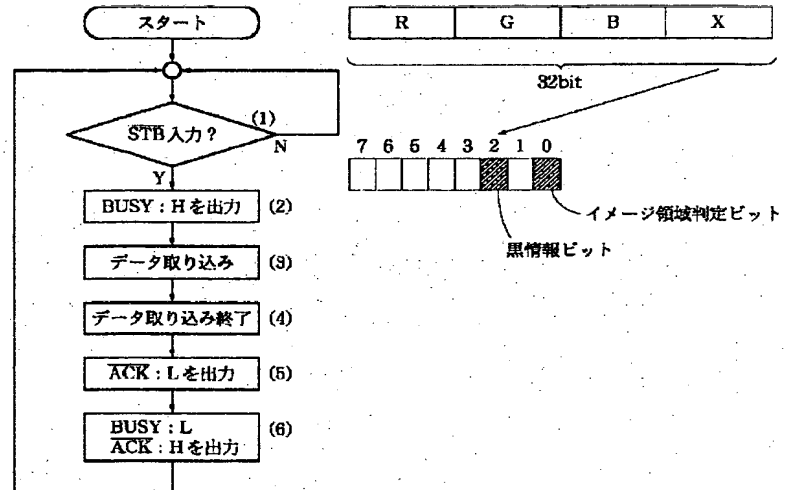
【图 14】



【図 18】

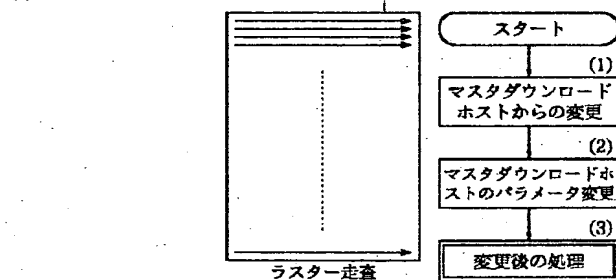


【図 24】

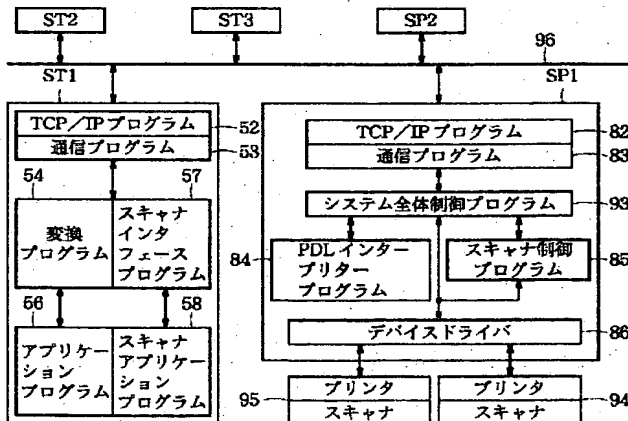
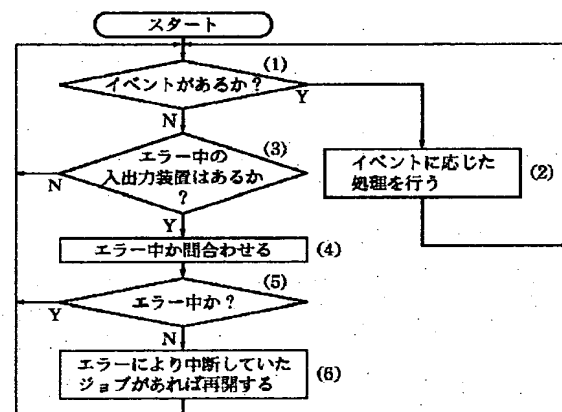


【図 83】

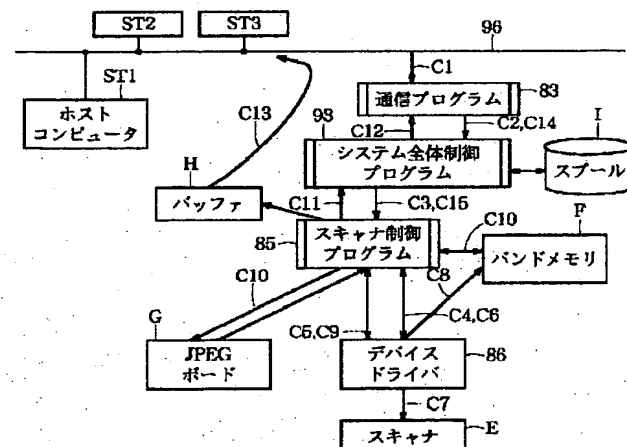
【図 26】



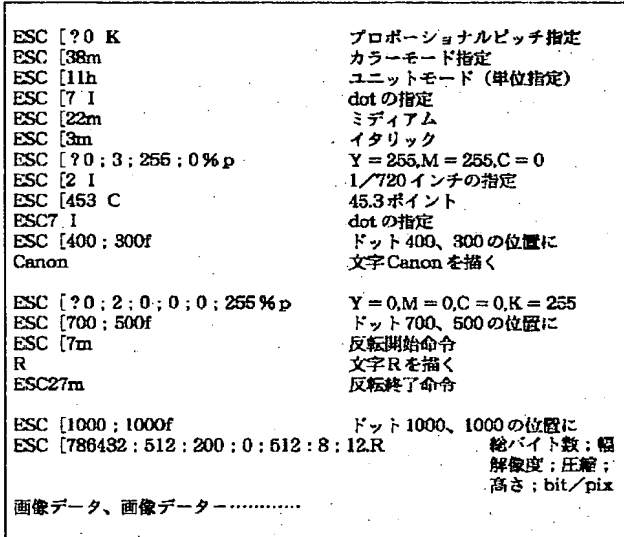
【図 25】



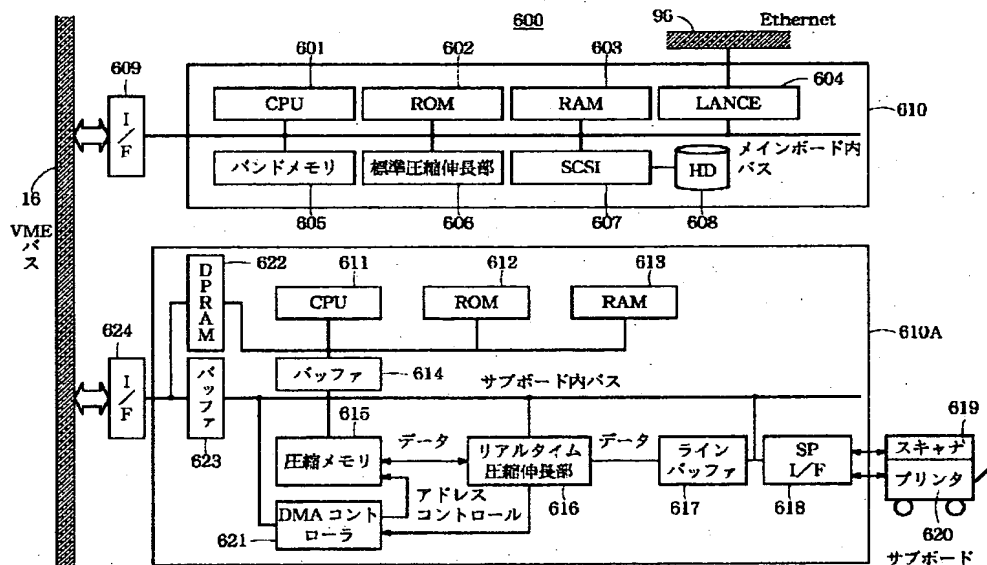
【図 31】



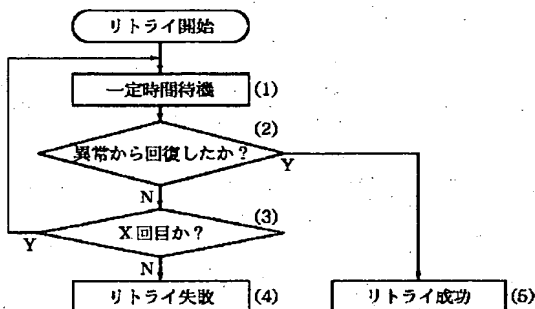
【図 27】



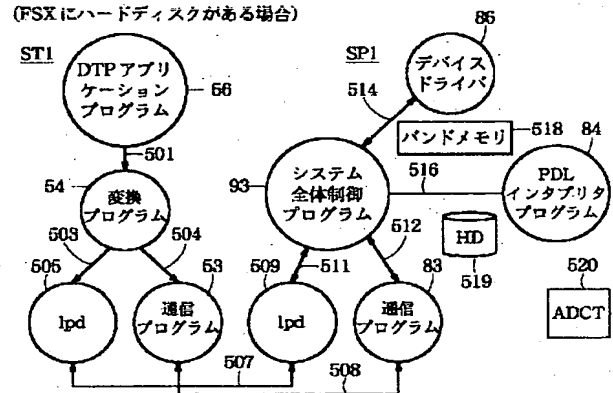
【図 33】



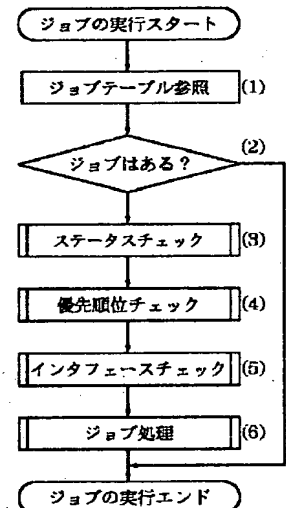
【図 37】



【図 32】



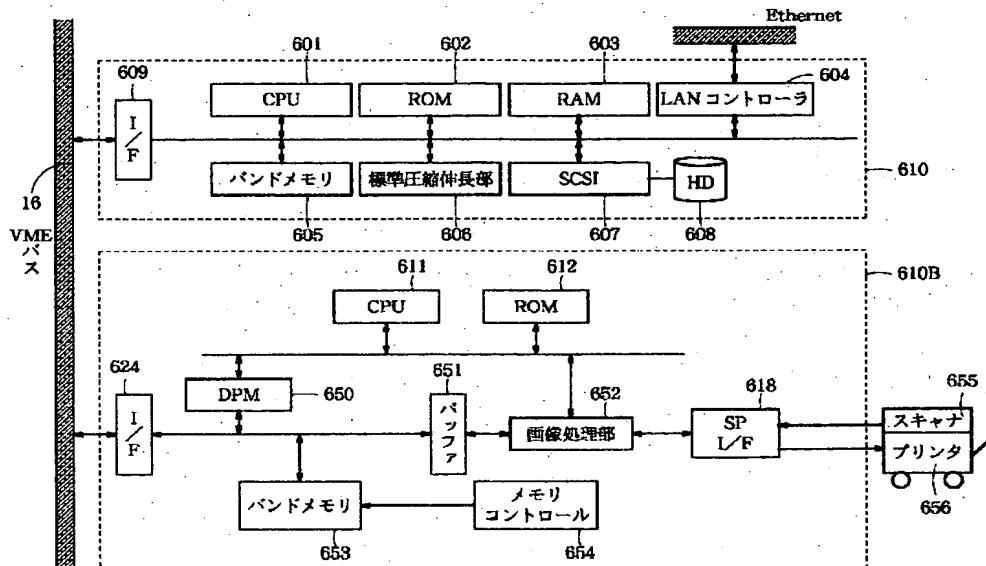
【図 42】



【図 41】

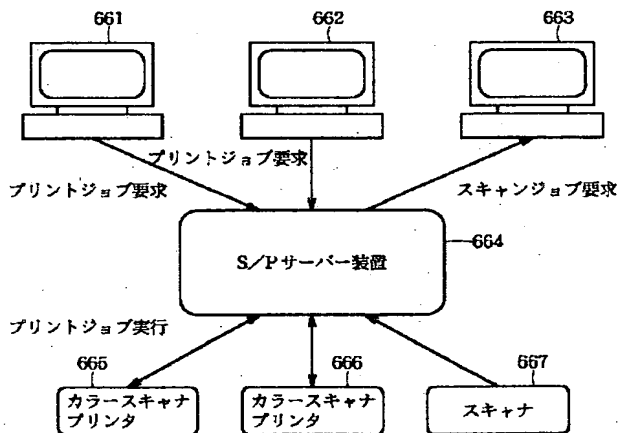
ID	ステータス	優先順位	インタフェース	ジョブ	終了ページ	割込ジョブID
25	ESTOP	LEVEL2	BMEM	CLCP	1	
26	STOP	LEVEL2	BMEM	BJP	2	27
27	RUN	LEVEL2	BMEM	BJS	1	
28	WAIT	LEVEL0	CNTR	PDLP	0	

【図 3 4】

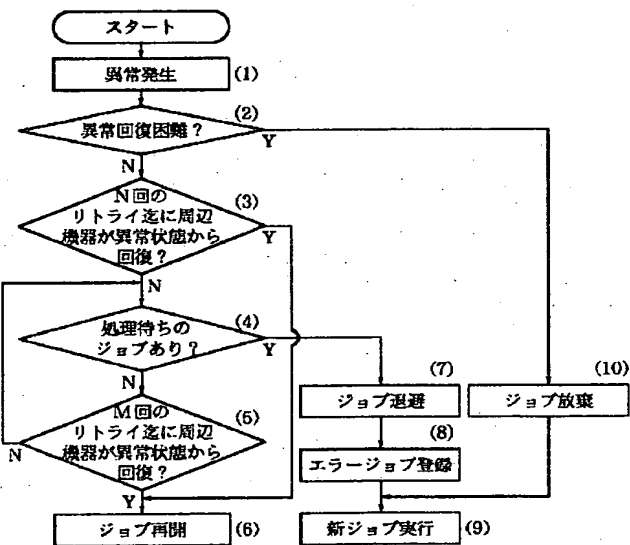


【図 3 5】

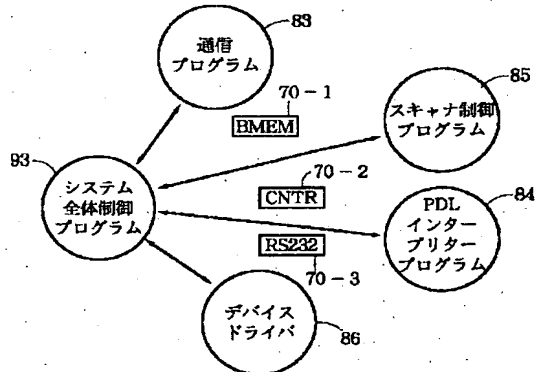
【図 3 6】



【図 3 9】

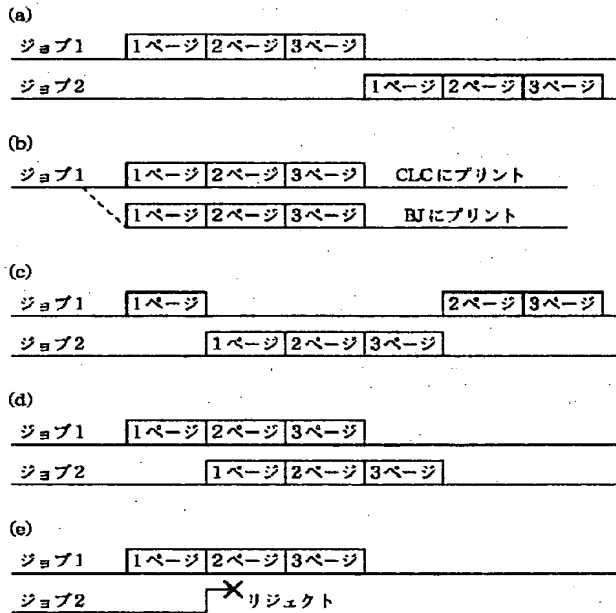


【図 4 8】

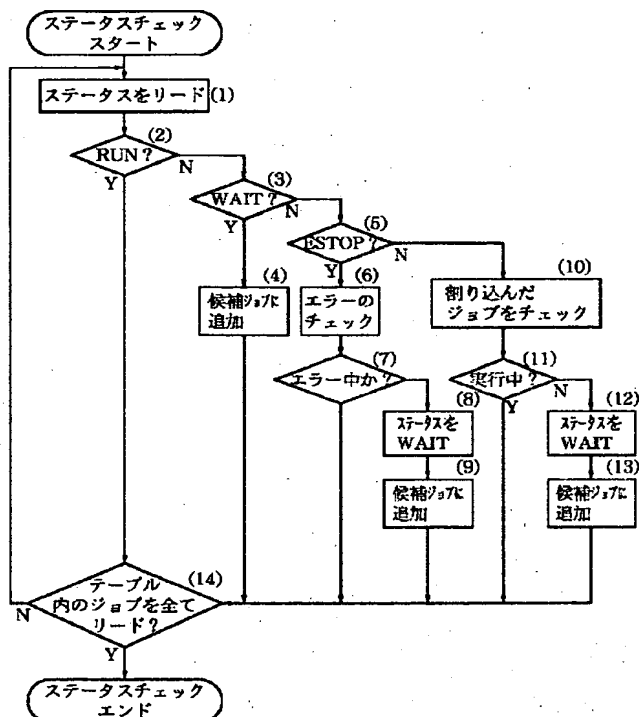


ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	到達ジョブID
25	WAIT	LEVEL0	BMEM	CLCP	0	

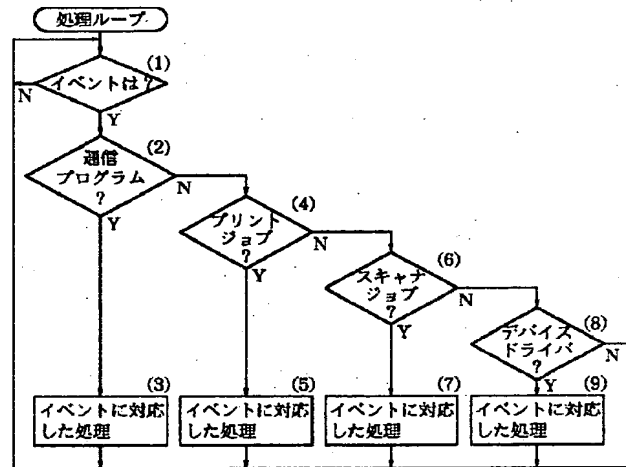
【図 38】



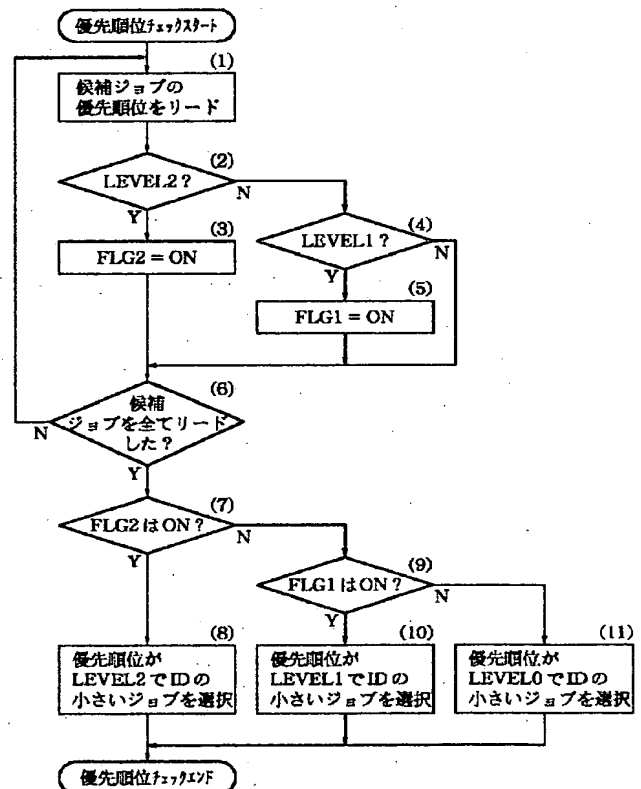
【図 43】



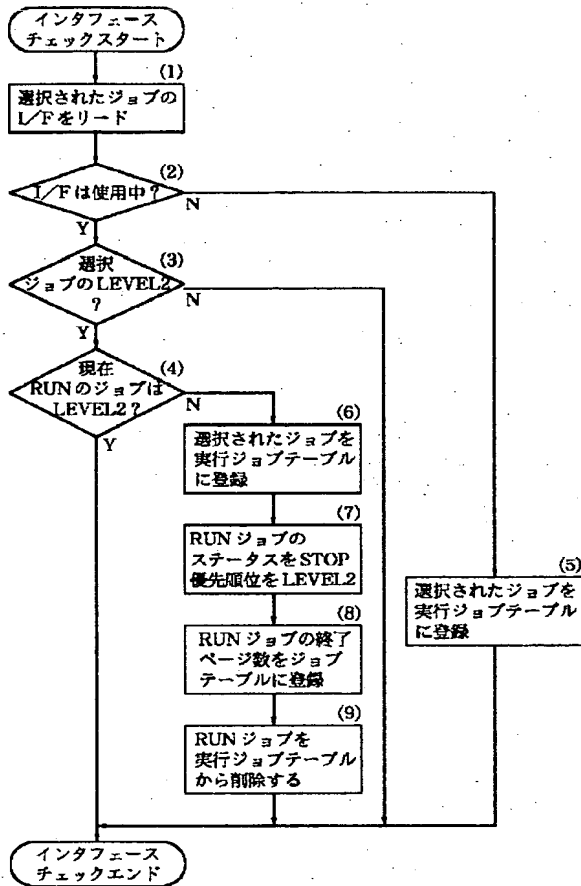
【図 40】



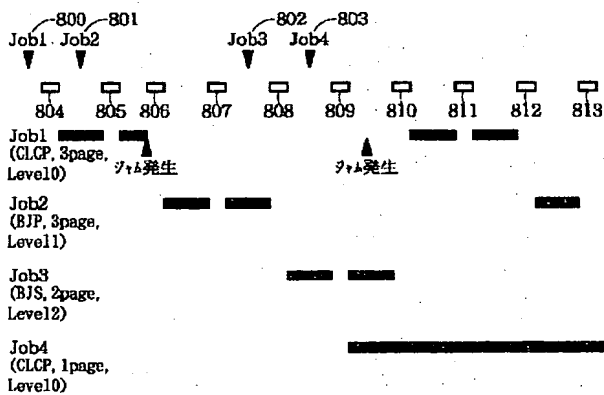
【図 44】



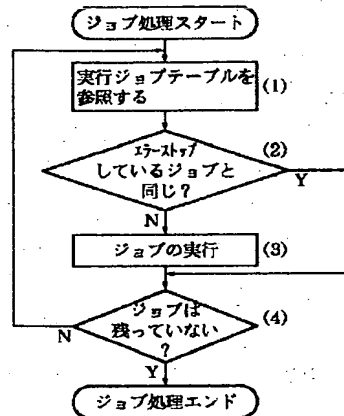
【図45】



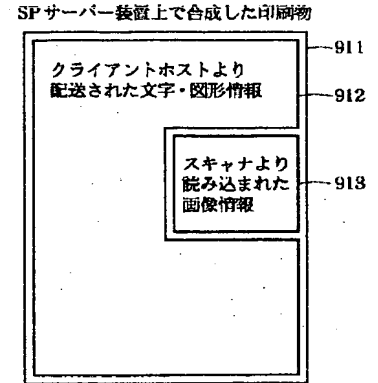
【図47】



【図46】



【図59】



【図49】

ID	ステータス	優先順位	インタフェース	ジョブ	終了ページ	割込ジョブID
25	RUN	LEVEL0	BMEM	CLCP	1	
26	WAIT	LEVEL1	BMEM	BJP	0	

【図50】

ID	ステータス	優先順位	インタフェース	ジョブ	終了ページ	割込ジョブID
25	ESTOP	LEVEL2	BMEM	CLCP	1	
26	WAIT	LEVEL1	BMEM	BJP	0	

【図51】

ID	ステータス	優先順位	インタフェース	ジョブ	終了ページ	割込ジョブID
25	ESTOP	LEVEL2	BMEM	CLCP	1	
26	RUN	LEVEL1	BMEM	BJP	1	

【図52】

ID	ステータス	優先順位	インタフェース	ジョブ	終了ページ	割込ジョブID
25	ESTOP	LEVEL2	BMEM	CLCP	1	
26	RUN	LEVEL1	BMEM	BJP	2	
27	WAIT	LEVEL2	BMEM	BJS	0	

【図 53】

ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	割込タプルID
25	ESTOP	LEVEL2	BMEM	CLCP	1	
26	STOP	LEVEL2	BMEM	BJP	2	27
27	RUN	LEVEL2	BMEM	BJS	1	
28	WAIT	LEVEL0	CNTR	PDLP	0	

【図 54】

ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	割込タプルID
25	WAIT	LEVEL2	BMEM	CLCP	1	
26	WAIT	LEVEL2	BMEM	BJP	2	
28	RUN	LEVEL0	CNTR	PDLP	0	

【図 55】

ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	割込タプルID
25	RUN	LEVEL2	BMEM	CLCP	2	
26	WAIT	LEVEL2	BMEM	BJP	2	
28	RUN	LEVEL0	CNTR	PDLP	0	

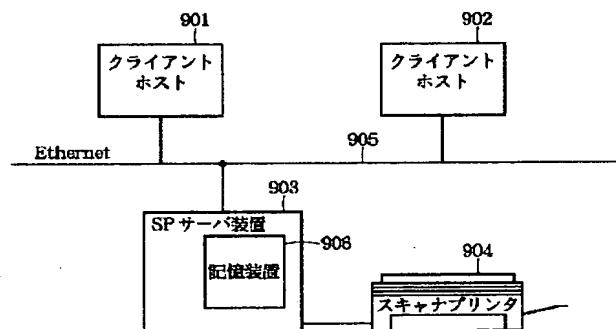
【図 56】

ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	割込タプルID
26	WAIT	LEVEL2	BMEM	BJP	2	
28	RUN	LEVEL0	CNTR	PDLP	0	

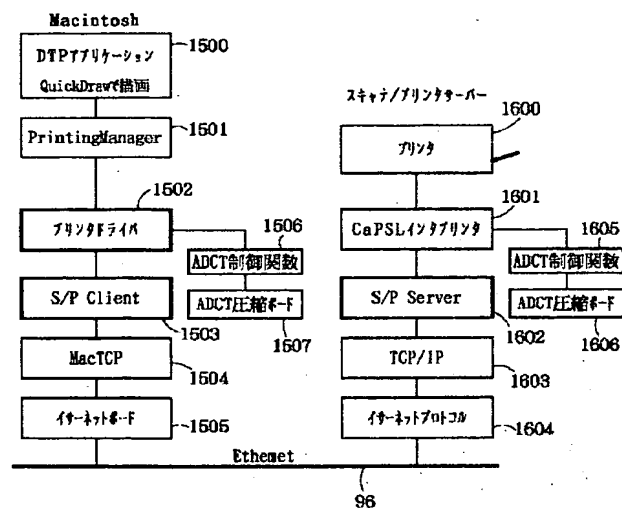
【図 57】

ID	ステータス	優先順位	インターフェース	ジョブ	終了ページ	割込タプルID
28	RUN	LEVEL0	CNTR	PDLP	0	

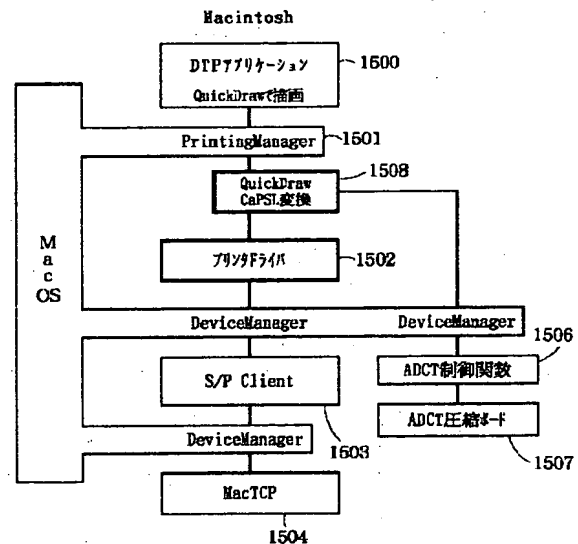
【図 58】



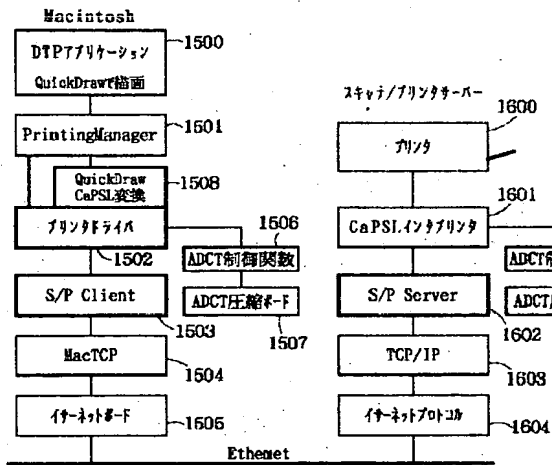
【図 60】



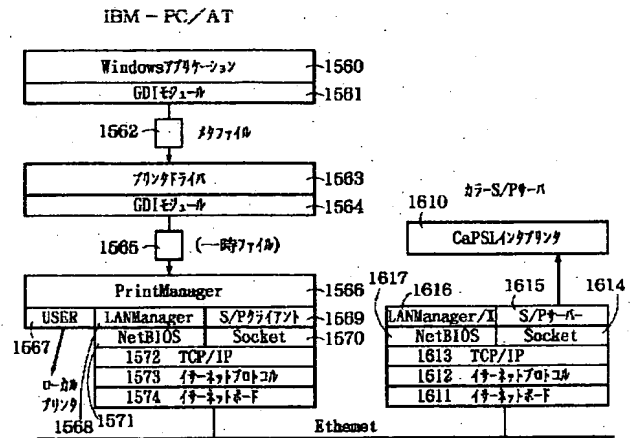
【図 61】



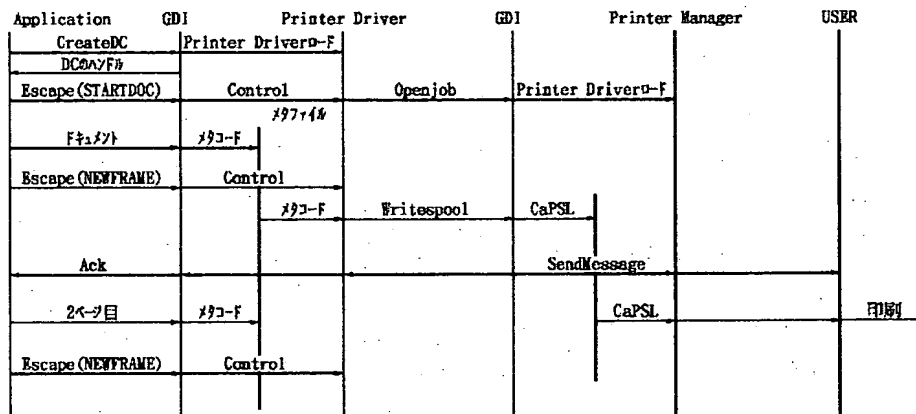
【図62】



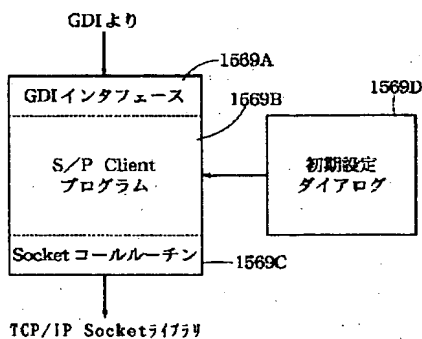
【図63】



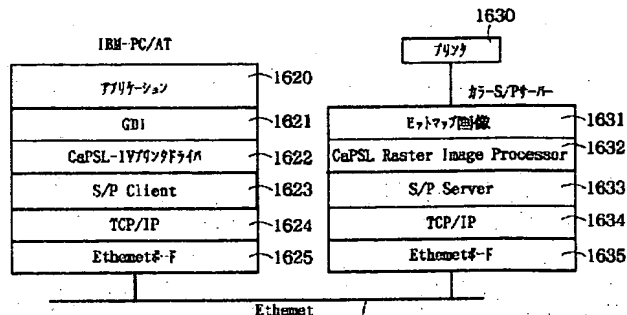
【図64】



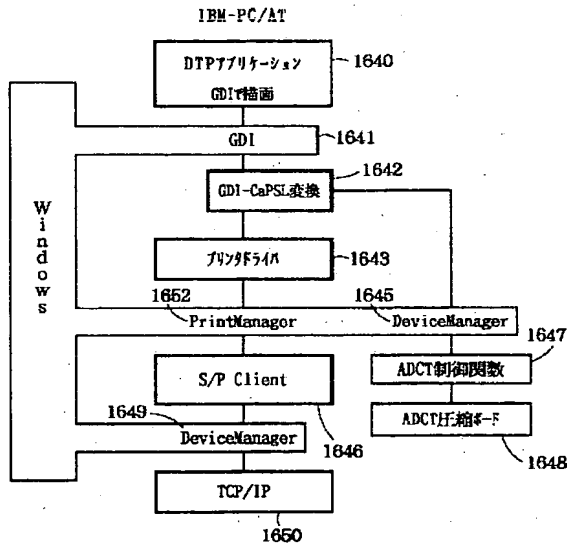
【図65】



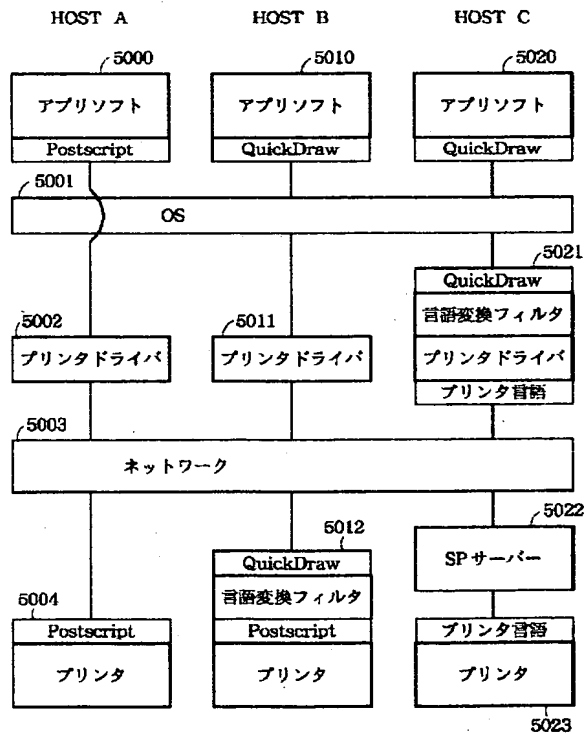
【図66】



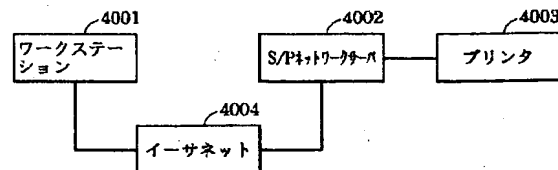
【図 67】



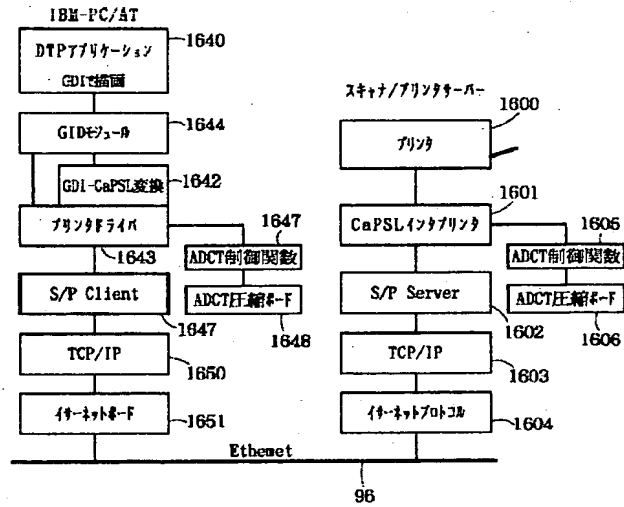
【図 69】



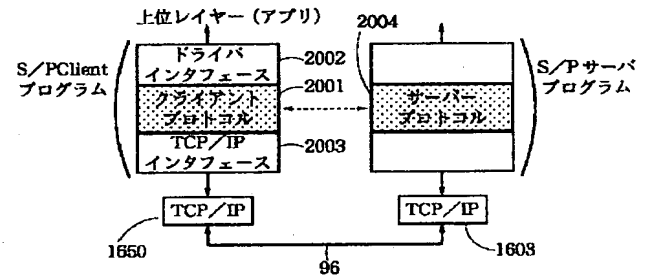
【図 75】



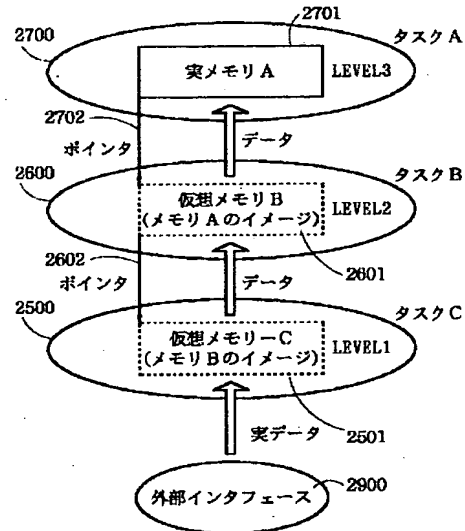
【図 68】



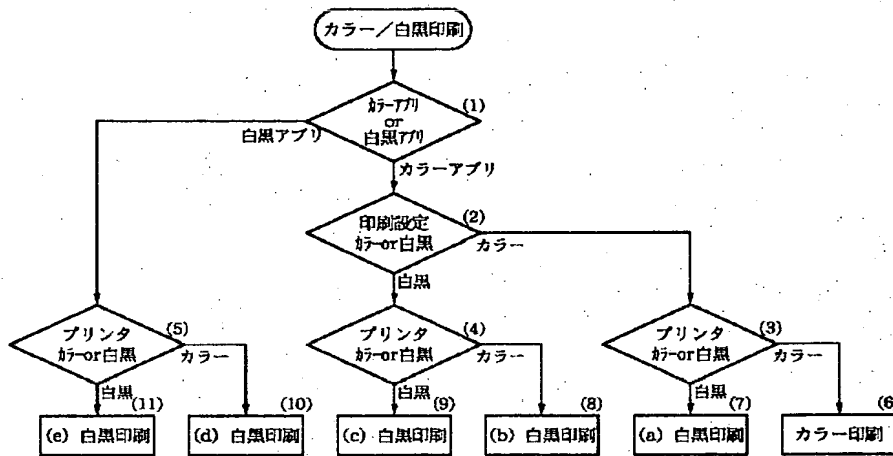
【図 70】



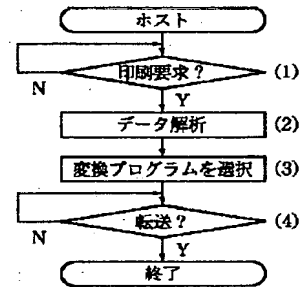
【図 71】



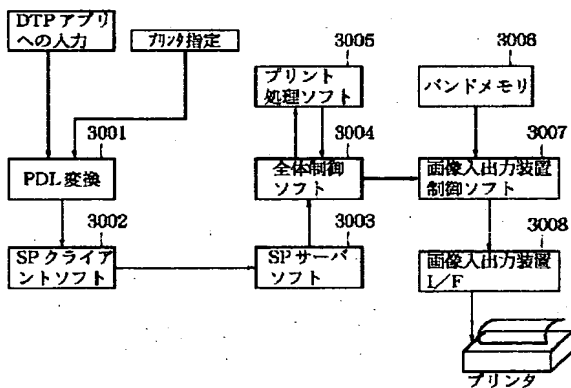
【図 7 2】



【図 9 2】



【図 7 3】

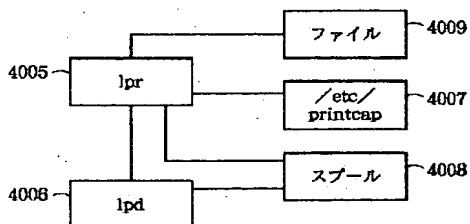


【図 7 4】

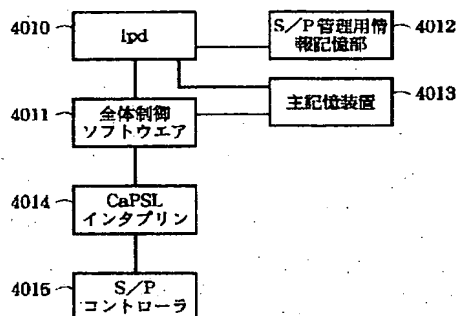
	PDL変換ソフト 3001			プリント処理ソフト 3005			画像入出力装置制御ソフト 3007		
条件	a	b	c	a	b	c	a	b	c
①	○	/	○	/	○	/	/	○	/
②	○	/	/	/	○	○	/	○	/
③	/	/	/	/	/	/	/	/	/
④	○	/	/	/	○	○	/	○	/
⑤	/	/	/	/	/	/	/	/	/

注
 ・○は処理実行箇所
 ・(c),(a)は正常処理
 ・条件(a)の時はプリント処理ソフト3005、画像入出力制御ソフト3007のどちらか一方で処理を行う。

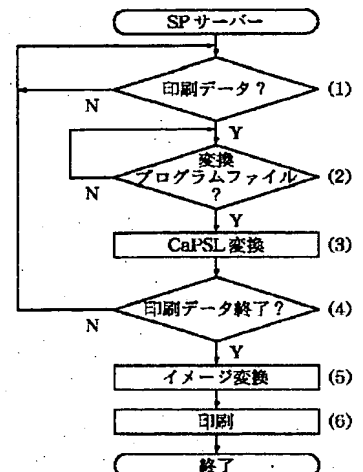
【図 7 6】



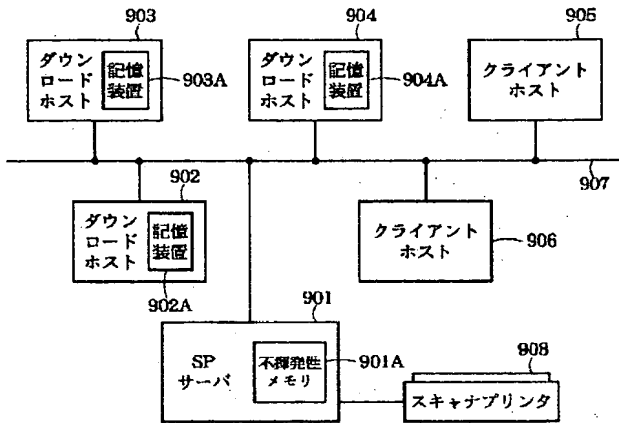
【図 7 7】



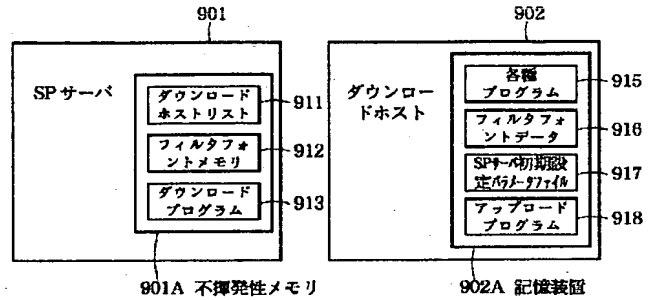
【図 9 3】



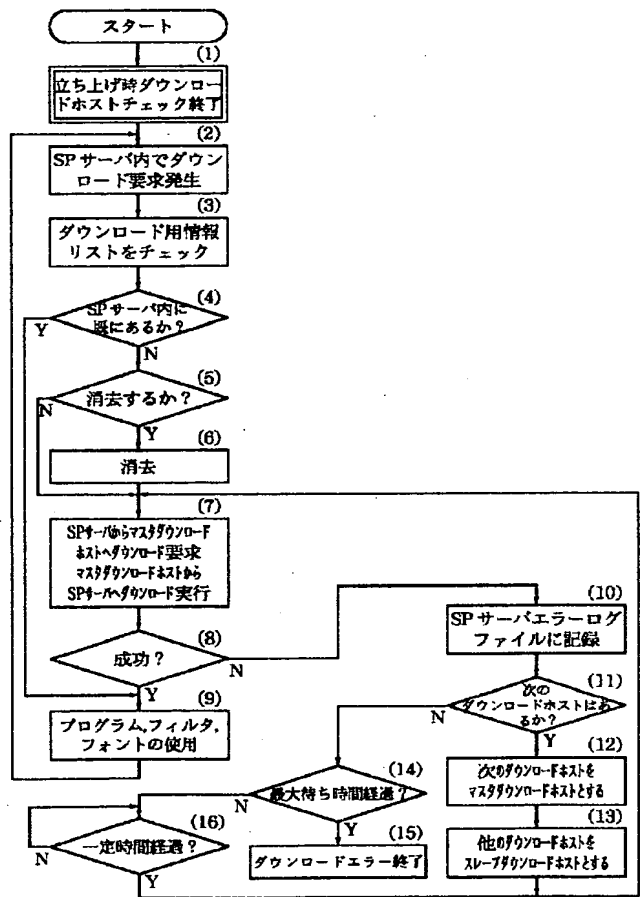
【図 78】



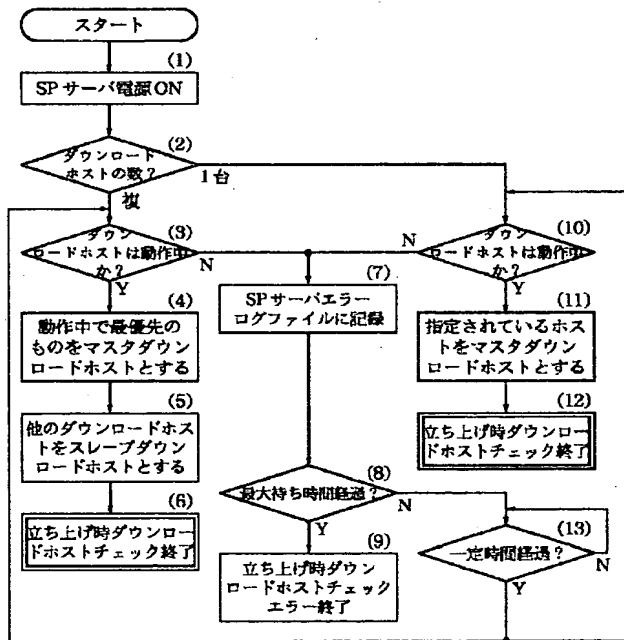
【図 79】



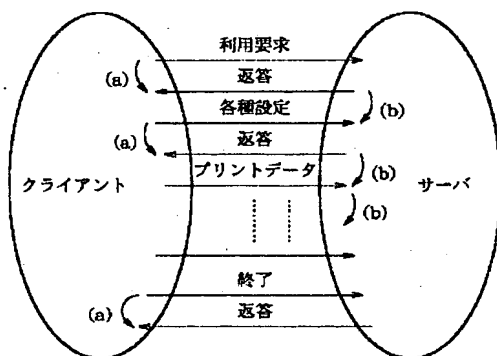
【図 81】



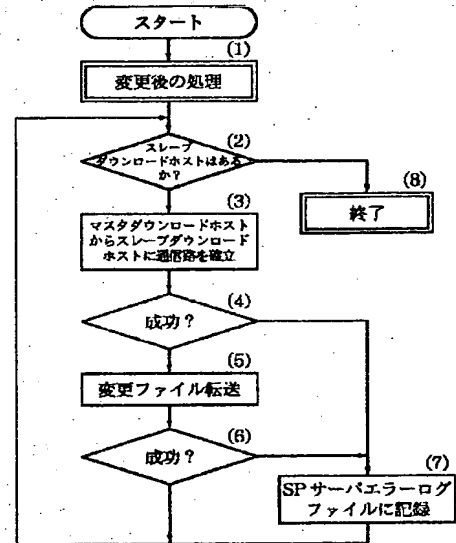
【図 80】



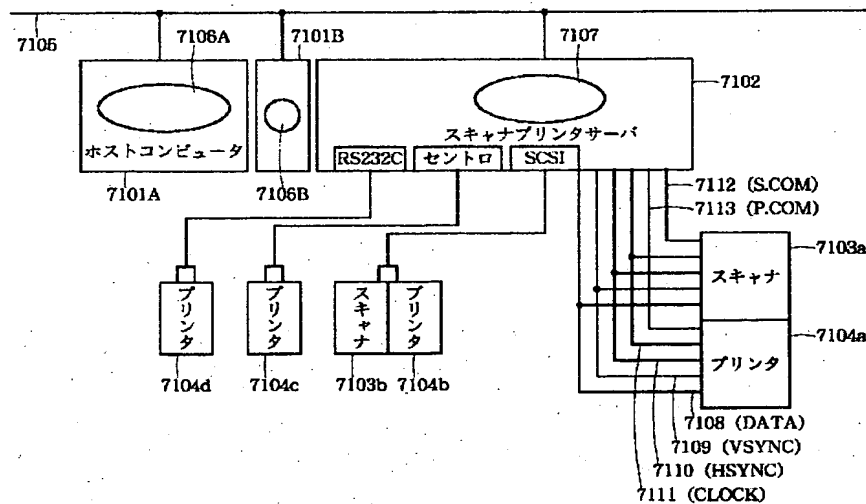
【図 86】



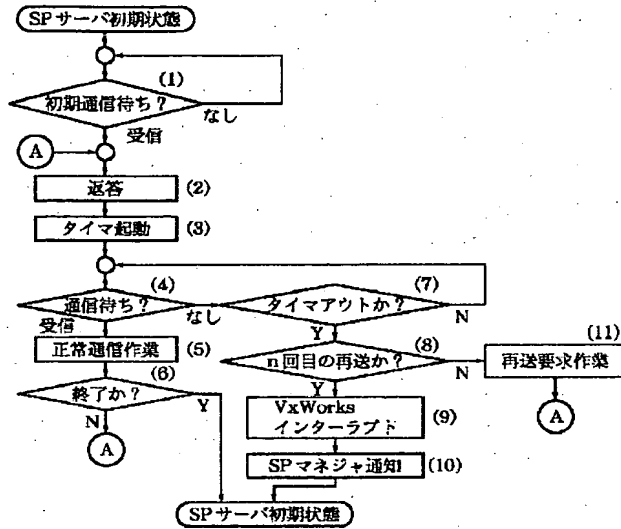
【圖 8 4】



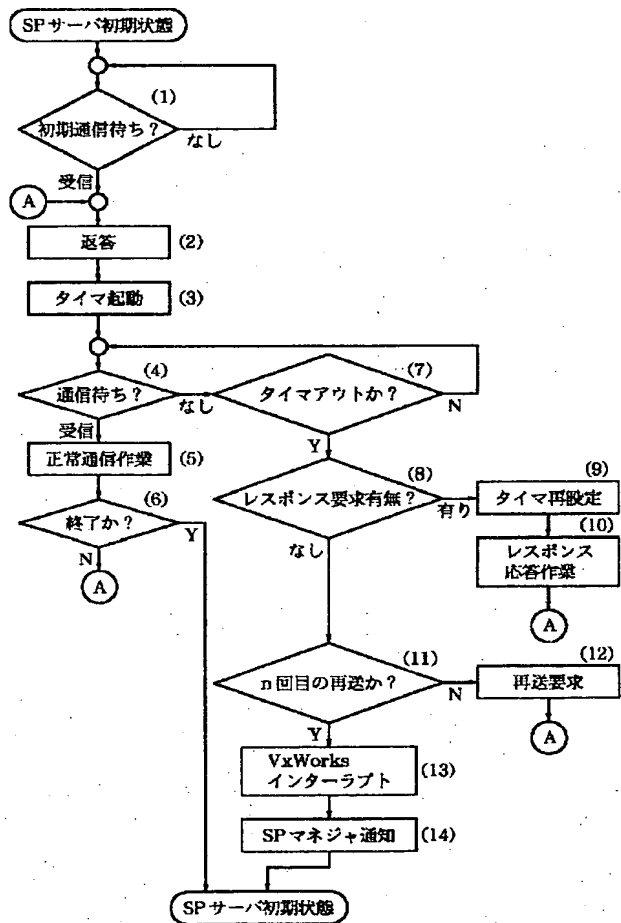
【図 85】



【図87】

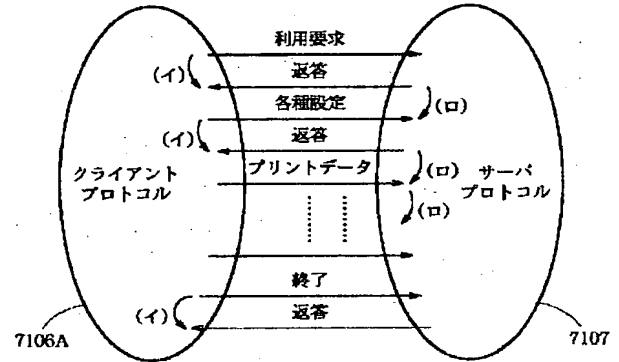


【図89】

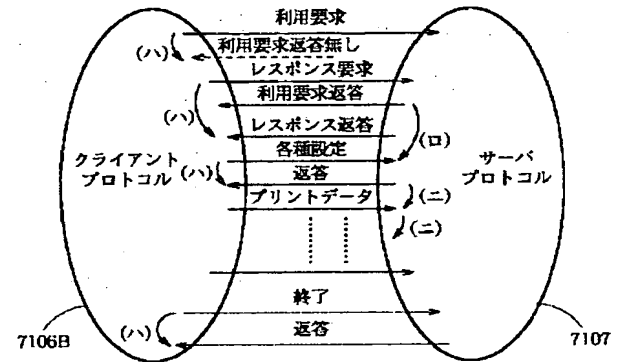


【図88】

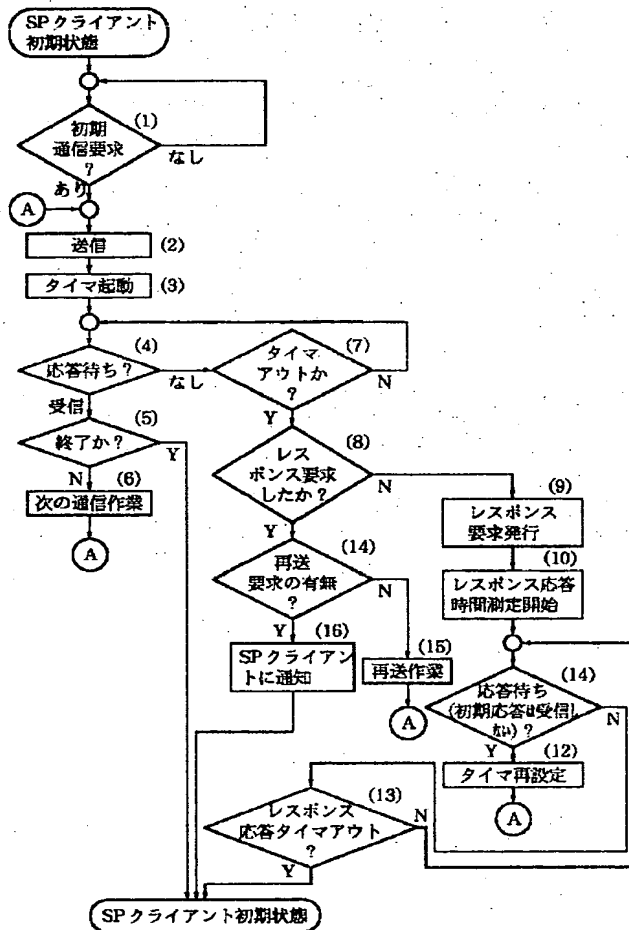
(a) 正常な通信



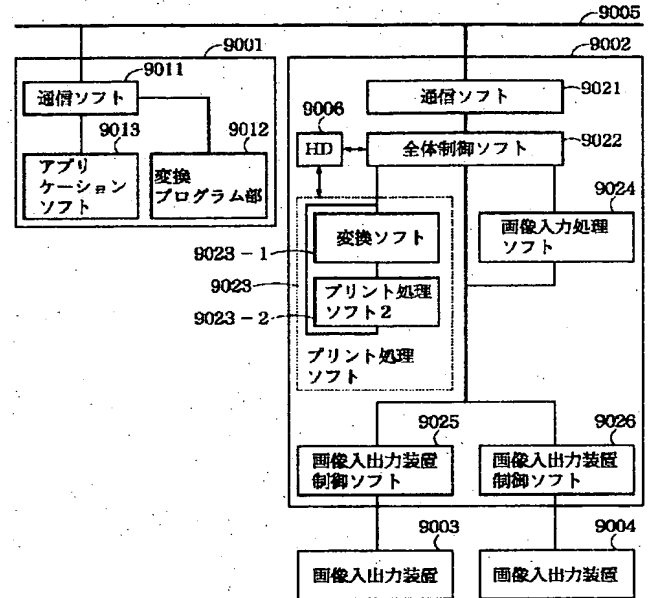
(b) レスポンスの遅い通信



【図90】



【図91】



9001 ホストマシン
9002 スキャナプリンタサーバ
9005 ネットワーク

フロントページの続き

(72)発明者 戸田 ゆかり
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 齋藤 和浩
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 戸田 雅成
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 根岸 作力
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 福田 康男
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 杉山 光正
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 穴塚 順一
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 高岡 真琴
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 小林 重忠
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72)発明者 三田 良信
東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(72) 発明者 杉浦 進

東京都大田区下丸子 3 丁目 30 番 2 号 キヤ
ノン株式会社内